AD

AMSAA

TECHNICAL REPORT NO. 279
REVISED

A COMPUTER PROGRAM FOR ESTIMATION OF PARAMETERS OF THE
WEIBULL INTENSITY FUNCTION AND FOR THE CRAMER-VON MISES
GOODNESS OF FIT TEST

EDWARD F. BELBOT

APRIL 1980

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

U. S. ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY
ABERDEEN PROVING GROUND, MARYLAND

DISPOSITION

Destroy this report when no longer needed. Do not return it to the originator.

DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position unless so specified by other official documentation.

WARNING

Information and data contained in this document are based on the input available at the time of preparation. The results may be subject to change and should not be construed as representing the DARCOM position unless so specified.

TRADE NAMES

The use of trade names in this report does not constitute an official endorsement or approval of the use of such commercial hardware or software. The report may not be cited for purposes of advertisement.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Technical Report No. 279 (Revised)	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Computer Program for Estimation of Parameters of the Weibull Intensity Function and for the Cramer-Von Mises Goodness of Fit Test		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Edward F. Belbot		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS U S Army Materiel Systems Analysis Activity Aberdeen Proving Ground, MD 21005		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS DA Project No. 1R765706M541
11. CONTROLLING OFFICE NAME AND ADDRESS Commander, U S Army Materiel Development and Readiness Command, Alexandria, VA 22333		12. REPORT DATE April 1980
		13. NUMBER OF PAGES 76
14. MONITORING AGENCY NAME & ADDRESS(If different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, If different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Weibull Intensity Function
Goodness of Fit Test
FORTRAN
Iterative Procedure

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the structure and use of a digital computer program written in Standard FORTRAN, which can be used to obtain the maximum likelihood estimates of the parameters of the Weibull intensity function. The program also performs the Cramer-Von Mises goodness of fit test.

CONTENTS

i

LIST OF FIGURES

A COMPUTER PROGRAM FOR ESTIMATION OF PARAMETERS FOR THE WEIBULL INTENSITY FUNCTION AND FOR THE CRAMER-VON MISES GOODNESS OF FIT TEST

1. INTRODUCTION

The Weibull intensity function

$$U(X) = \lambda\beta X^{\beta-1} \qquad (1.1)$$

$\lambda > 0$, $\beta > 0$, $X > 0$, is frequently used as a model for the determination of reliability growth and wear-out characteristics for a wide variety of complex, repairable systems. The failure rates of military equipment such as vehicles, aircraft, guided missiles, electronic computer systems, and ammunition are being evaluated using this model.

Formulas have been developed by the Army Materiel Systems Analysis Activity (AMSAA) for the maximum likelihood estimation of the unknown parameters λ and β, based upon sample data. These estimation formulas for the Weibull process, found in Crow (1975), can be stated as

$$\hat{\beta} = \frac{\sum\limits_{i=1}^{K} N_i}{\hat{\lambda} \sum\limits_{i=1}^{K} [T_{i2}^{\hat{\beta}} \ln(T_{i2}) - T_{i1}^{\hat{\beta}} \ln(T_{i1})] - \sum\limits_{i=1}^{K} \sum\limits_{j=1}^{N_i} \ln(X_{ij})} \qquad (1.2)$$

$$\hat{\lambda} = \frac{\sum\limits_{i=1}^{K} N_i}{\sum\limits_{i=1}^{\hat{K}} (T_{i2}^{\hat{\beta}} - T_{i1}^{\hat{\beta}})} \qquad (1.3)$$

where:

  K is the number of systems under study;

  N_i is the total number of failures (or occurrences of an event under study, such as unscheduled maintenance actions; etc.) for the ith system;

  T_{i1} is the starting time of the period of continuous observation of the ith system;

  T_{i2} is the ending time of the period of continuous observation of the ith system;

  X_{ij} is the jth time of occurrence of the failure (or event), for the ith system;

ln is the natural logarithm, and $0 \cdot \ln(0)$ is defined to be 0.

To expedite the computation of these estimates, AMSAA developed a FORTRAN computer program to calculate $\hat{\beta}$ and $\hat{\lambda}$. That program, which was documented in Belbot (1974), was successfully employed by the U S Army Materiel Development and Readiness Command (DARCOM), various subordinate commands and several project managers' offices, as well as by AMSAA. After the parameters were determined by that program, a goodness of fit test was frequently used to test statistically the hypothesis that the failure times of the systems being analyzed followed a nonhomogeneous Poisson process with Weibull intensity function (see Crow[1975]). The modified Cramér-Von Mises goodness of fit statistic was computed, either by hand or by a separate computer program.

Obviously, the consolidation of an automated goodness of fit test with the computer routine which estimates the parameters $\hat{\beta}$ and $\hat{\lambda}$, would increase efficiency and accuracy. Either the manual calculations or the use of a separate program would be eliminated. To accomplish properly this consolidation, the estimation procedure was subordinated to a new main program which also controls the input of data and the execution of the goodness of fit test. Because of the radical nature of this redesign, it was appropriate to incorporate other new features at the same time. Principal among these new features are a simplified input procedure and dynamic data storage allocation. The resulting computer program is easier to use and provides more information that its antecedent program. This report will explain the structure and the use of this new program.

2. COMPUTING PROCEDURE

2.1 Estimation of Parameters

Since the formulas (1.2) and (1.3) do not, in general, yield $\hat{\beta}$ and $\hat{\lambda}$ in closed form, an iterative technique is required. Formula (1.2) may be recast as

$$\frac{\sum\limits_{i=1}^{K}\sum\limits_{j=1}^{N_i} \ln(X_{ij})}{\sum\limits_{i=1}^{K} N_i} - \frac{\sum\limits_{i=1}^{K}[T_{i2}^{\hat{\beta}}\ln(T_{i2})-T_{i1}^{\hat{\beta}}\ln(T_{i1})]}{\sum\limits_{i=1}^{K}(T_{i2}^{\hat{\beta}}-T_{i1}^{\hat{\beta}})} - \frac{1}{\hat{\beta}} = 0 \qquad (2.1)$$

by replacing $\hat{\lambda}$ by its equivalent expression from equation (1.3), and by execution of a few simple algebraic operations. Equation (2.1) now consists of a constant with regard to $\hat{\beta}$, minus a function of $\hat{\beta}$, yielding 0, or simply

$$A - D(\hat{\beta}) = 0 \qquad (2.2)$$

2

The correct value of $\hat{\beta}$ will satisfy equation (2.2) and can be used to calculate the corresponding value of $\hat{\lambda}$.

The solution for $\hat{\beta}'$ is iteratively determined in the following way. For an initial estimate $\hat{\beta}'$ which is assumed to be larger than the true $\hat{\beta}$, the expression A-D($\hat{\beta}'$) is evaluated. For all values of $\hat{\beta}'$ larger than the true $\hat{\beta}$, the subtraction yields a negative result. After each negative result, $\hat{\beta}'$ is reduced by the initial step size of 1, and A-D($\hat{\beta}'$) is again evaluated.

When a positive number results from the subtraction, indicating that $\hat{\beta}'$ is too small, the step size is decreased to 0.10 of the present step size, the previous value of $\hat{\beta}'$ which gave a negative result for A-D($\hat{\beta}'$), is adjusted by the new step size and the evaluation process begins again.

The iteration procedure continues, adjusting $\hat{\beta}'$ by the new step sizes, until the left side of equation (2.2) is within a specified tolerance ϵ of 0. $\hat{\lambda}$ is then calculated based on $\hat{\beta}$, using factors already computed in finding $\hat{\beta}$. This procedure is summarized by the state diagram (Figure 2.1).

2.2 Goodness of Fit Test

The Cramer-Von Mises Goodness of Fit Test is appropriate whenever the starting time of each system is equal to 0. To perform this test, the program first transforms the failure times. For time truncated testing, the failure times for each system are divided by the ending time of the test period for that system. In failure truncated testing, for every system, all the failure times except the last, are divided by the last failure time. The last failure time is thereafter excluded from the calculations and the number of transformed failures is one less than the original number of failures for each such system. All the transformed failure times are then sorted into increasing order.

Next, the unbiased estimate $\overline{\beta}$ of the estimated shape parameter $\hat{\beta}$, is calculated using the relation:

$$\overline{\beta} = \frac{M-1}{\sum_{i=1}^{K} L_i \ln(T_{i2}) - \sum_{i=1}^{K} \sum_{j=1}^{L_i} \ln(X_{ij})} \qquad (2.3)$$

where:

$$L_i = \begin{cases} N_i-1 & \text{if the data for the } i^{th} \text{ system are failure truncated} \\ N_i & \text{if the data for the } i^{th} \text{ system are time truncated} \end{cases}$$

3
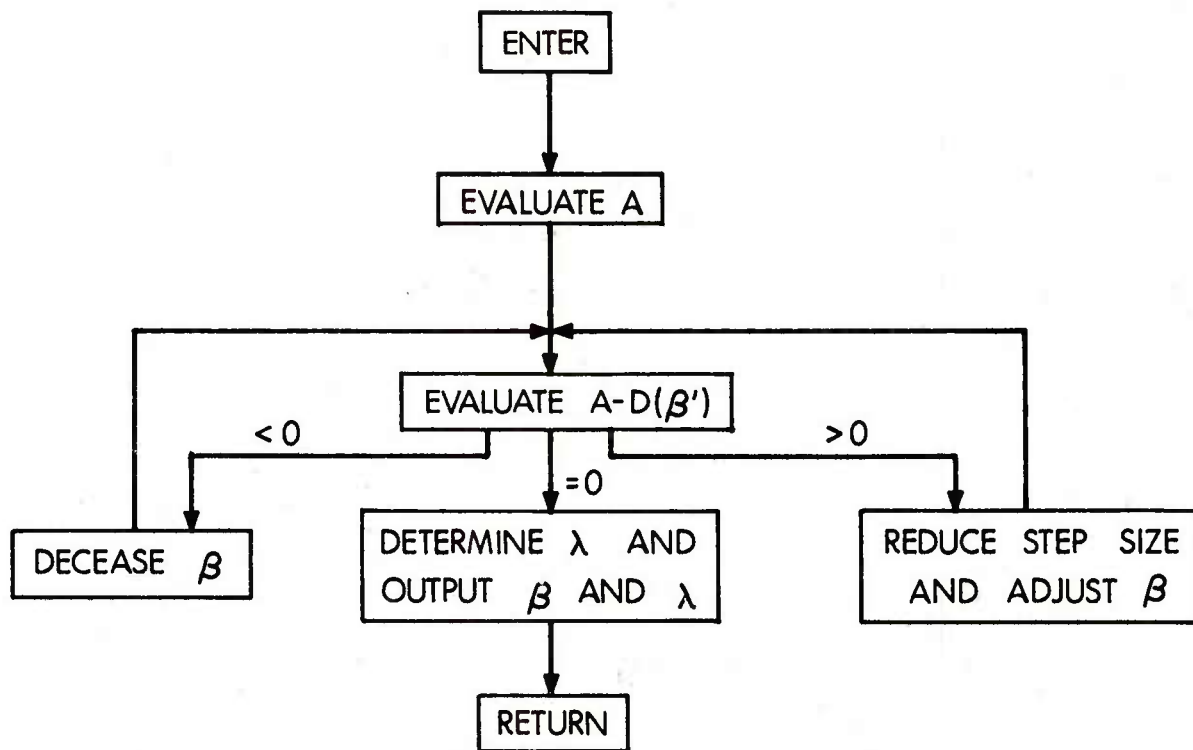
Figure 2.1   State   Diagram

4

and M = total number of transformed failure times. (Note that

$$M = \sum_{i=1}^{K} L_i .)$$

Finally, the Cramer-Von Mises statistic C_M^2 is computed by the formula:

$$C_M^2 = \frac{1}{12M} + \sum_{i=1}^{M} (Z_i^{\overline{\beta}} - \frac{2i-1}{2M})^2 \qquad (2.4)$$

where the Z_i are the transformed failure times. An explanation of this test and a table of critical values of C_M^2 may be found in Crow (1975).

Next page is blank.

3. DESCRIPTION OF PROGRAM

3.1 Major Features

The program, which is listed in Appendix A, has some important features. First, the program is written in American National Standards Institute (ANSI) FORTRAN X3.9-1966, and should therefore execute on any computer having a compiler for this language. Secondly, the amount of storage required to use this program should not cause difficulties since all data arrays are dynamically allocated under control of the main program (see Chung-Phillips, et al., 1975).

Finally, and perhaps most importantly for the user, this program uses free-field input, that is, no specific format is required for the input data. The time and the effort regularly expended in preparing data for input, are greatly reduced because of this feature. Moreover, the misalignment of data fields to formats, a frequent source of errors in using many computer programs, is eliminated entirely. While no input scheme can be regarded as foolproof, free-field input is much more flexible than fixed-field format specifications.

3.2 Overall Characteristics

All calculations in the program are made in double precision mode. Experience has shown that the use of single precision variables for these calculations often results in significant discrepancies in the estimates of the parameters due to errors accumulated during the iterative process.

The modular structure of the program (see Figure 3.1) reflects organization by functional purpose. The input of data, certain intermediate calculations, the estimation of parameters, and the goodness of fit test are each performed by an independent module. Major subprograms print their results as the values become available. Subroutines which detect errors, print diagnostic messages naming the detecting routine and briefly stating the difficulty, and then attempt to continue processing when possible. Independence of the subroutines is maintained by restricting communication between individual subprograms to the passage of formal parameters in argument lists. No COMMON statements are used.

3.3 Specific Details of Routines

In addition to controlling the major modules, the main program also allocates storage for data arrays, as stated in Section 3.1. The allocation is based upon the maximum number of failures, NFAIL, and the maximum number of systems, NSYS. The master data array, BLOCK, has a length of NTOTAL, equal to 1.5 times the value of NFAIL plus six times the value of NSYS. To redimension the entire program, one merely adjusts the values of NFAIL, NSYS and NTOTAL in the DATA statements at the beginning of the main routine and changes the size of the array BLOCK, also found at the beginning of the main program, to equal the new value of NTOTAL.

7
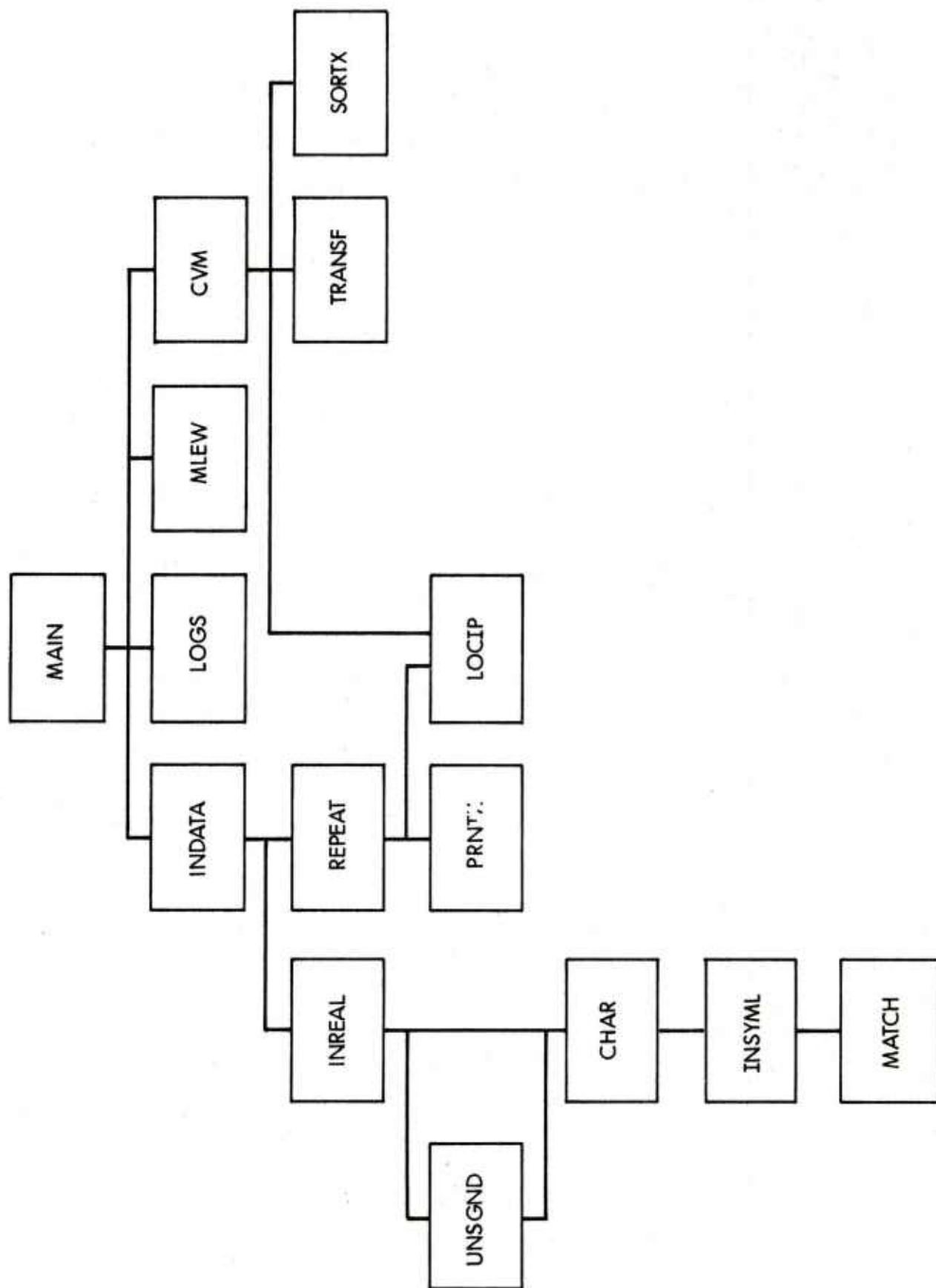
Figure 3.1 Module Organization Chart.

By these actions, all data arrays in all subroutines will be properly resized. Since the program size excluding data arrays, is less than 5,000 words, computer memory requirements can be scaled to problem size through use of this feature.

Also found in the DATA statements at the beginning of the main routine are the unit number for input, IUNIT, and the logical switch ECHO which controls the printing of the input data. Just as the storage allocation values, these values may be changed as needed.

The first major module is for the input of data. The INDATA module reads the beginning and the ending times and the failure times from the input unit, IUNIT. If the logical variable ECHO is true, the submodule REPEAT will print the input, using the LOCIP subroutine to isolate in storage the failures for each system and the PRNTX subroutines to print them. The failure times are stored in a linear array called X. Parallel to X, an integer array named MARKER identifies the failures with the respective systems, such that if failure X(I) occurred on system J, then MARKER(I) equals J.

The free-field reading of data is performed by the INREAL sub-module. This submodule, consisting of the routines INREAL, CHAR, INSYML, MATCH and UNSGND, is a translation from ALGOL into FORTRAN of Algorithm 239 of the Association for Computing Machinery (see McKeeman [1964]). Specific details concerning the input arrangement are given in Section 4.

The INDATA subroutine also sets three logical variables depending on the input. If data errors are encountered, the variable NOGOOD is made true. If all systems start at 0.0, then the goodness of fit test will be appropriate and so the variable GOF is set to true. Lastly, when the end of input is reached, the logical variable HALT is returned as true.

The second principal module, LOGS, calculates the logarithms of the beginning and the ending times, and the sum of the logarithms of the failure times. For computational purposes, a beginning time of 0.0 is defined to have a logarithm of 0.0 instead of infinity. Note that failure times of 0.0 are not valid for this model and are flagged as errors by the preceding INDATA module.

The next major module, MLEW, computes the maximum likelihood estimates of the parameters of the Weibull intensity function for the given data, using the formulas discussed in Section 2.1. If unsuccessful, this module will report one of three possible error conditions. The first error message "BETA LESS THAN 0.00000001" indicates that the data should be rechecked for the reasonableness of a very small $\hat{\beta}$. The second error message is "INITIAL ESTIMATE OF BETA IS TOO SMALL." Since the initial estimate of BETA is set to 10 at the start of the MLEW sub-program, this message indicates some pecularity of the data. (In general, $0 < \hat{\beta} < 10$.) The third message, "STEP-SIZE HAS BECOME INSIGNIFICANT - BETA NOT RESOLVABLE," indicates that the module has gone as far as

9

possible trying to meet the tolerance set for the difference A-D($\hat{\beta}'$).
This tolerance, EPSILN, may be enlarged by changing the assignment state-
ment also located at the beginning of the MLEW subprogram.

The last major module, CVM, performs the Cramér-Von Mises goodness
of fit test, as described in Section 2.2. The failure times for each
system are located in storage using the LOCIP subroutine, and examined to
determine if the testing was time truncated or failure truncated. The
failure times are then transformed by the TRANSF subroutine and sorted by
the SORTX subroutine. (SORTX is a modification of an utility subprogram
described in Campbell, et al., [1970].) The unbiased estimate of β, UNBETA,
is calculated next, as explained earlier. The last phase depends on the
system starting times. If any starting time is non-zero, the module

terminates with a message stating that the Cramér-Von Mises goodness of
fit test is not appropriate. Otherwise, the goodness of fit statistic,
CM2, is computed and printed.

4. INPUT REQUIREMENTS

As stated previously, the input for this program is free-field. The only requirement regarding spacing is that at least one blank column separate adjacent values. The values must not run together. This means that the program generally takes the same view of the data that a person would, namely, that each cluster of numeric characters constitutes one data value. The only exception to this rule occurs at the boundaries of records. Since the input is treated as a continuous stream, a string of characters beginning in the first column of a record, is considered a continuation of the string of characters ending in the last column of the previous record, if any. Record boundaries are not delimiters; blanks are the only delimiters.

The data required for this program consist of the beginning and ending times of the test period for each system, and the failure times for each system. The arrangement of the input, (which is also stated in the comments of the INDATA subroutine), is by system. The first data value is the beginning time of the first system. The second data value is the ending time of the first system. Next is the failure times for the first system, followed by a negative value to mark the end of the first system. The same pattern, beginning time, ending time, failure times, and negative trailer, is repeated for each subsequent system in the first data case.

Another negative value (making two in a row), signals the end of input of the current data case, and the beginning of the computational procedures. The same arrangement may be repeated for as many cases as desired per program run. When the input routine encounters a negative value after completing a case, (that is, the third negative value in a row), the end of the program run is indicated.

Thus, as a simple example, if one desired to use this program for one run consisting of one case wherein one system experienced seven failures, the input data would be: the beginning time, the ending time, the seven failure times, and three negative values. To demonstrate the latitude of the input requirements, the data for a number of test cases are shown in Appendix B. Notice that any negative value is acceptable as a trailer and that data values may be entered with or without decimal points. Although not shown in the examples, data values may also be in exponential form, that is, containing 'E,' '+' or '-.'

Next page is blank.

5. TRANSFERABILITY AND MODIFICATION

Since this program is written in standard FORTRAN, transfer to other computer systems should be straightforward. To assist in the transfer process, Appendix C contains the output produced by the program for the input shown in Appendix B. This output was generated on a Control Data Corporation (CDC) Cyber 76 Computer, using the program exactly as listed in Appendix A. (Note that non-standard PROGRAM statement required by the CDC Cyber.)

The input for these test cases came from records of eighty characters each. If the input record length is other than eighty, two changes may be required. The value of the variable LENGTH and, if necessary, the dimensioned size of the array BUFFER, should be adjusted in the INSYML subroutine.

Alternatively, one could replace the entire INREAL submodule. Although these routines were written to be fully transportable, running time might be saved by using the system defined free-field reading capability of any computer having such a feature. As an example, a substitute for the INREAL submodule, suitable for the CDC Cyber 76, is shown in Figure 5.1. Such substitutes, however, are system dependent and not readily transferable.

Next page is blank.

FIGURE 5.1   SUBSTITUTE FOR INREAL SUBMODULE


```
SUBROUTINE INREAL (IU, X)
DOUBLE PRECISION X
READ (IU, *) X
RETURN
END
```

Next page is blank.

REFERENCES

1. Belbot, E. F., A Computer Program for Estimation of Parameters of the Weibull System Failure Rate Function, U S Army Materiel Systems Analysis Activity Technical Report No. 99, 1974.

2. Campbell, L. W. and Beck, G. A., BRLESC I/II FORTRAN, U S Army Aberdeen Research and Development Center Technical Report No. 5, 1970.

3. Chung-Phillips, A. and Rosen, R. W., "A Note on Dynamic Data Storage in FORTRAN IV," Computer Journal, Volume 18, No. 4, 1975.

4. Crow, L. H., Reliability Analysis for Complex, Repairable Systems, U S Army Materiel Systems Analysis Activity Technical Report No. 138, 1975.

5. McKeeman, W. M., "Algorithm 239 - Free Field Read," Communications of the Association for Computing Machinery, Volume 8, 1964.

6. USA Standard FORTRAN (USAS X3.9 - 1966). USA Standards Institute (now designated the American National Standards Institute, Inc.), 1966.

Next page is blank.

APPENDIX A

LISTING OF PROGRAM

Next page is blank.

```
      PROGRAM MAIN (INPUT=/80,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)          MAIN   10
C     THE ABOVE STATEMENT IS NON-STANDARD, BUT REQUIRED FOR            MAIN   20
C     CDC FORTRAN.                                                     MAIN   30
C                                                                      MAIN   40
C     MAIN DRIVER FOR WEIBULL INTENSITY MODEL PARAMETER                MAIN   50
C     ESTIMATION AND GOODNESS OF FIT TEST.  (12 NOV. 1979)             MAIN   60
C     INPUT REQUIREMENTS ARE DESCRIBED IN THE 'INDATA'                 MAIN   70
C     SUBROUTINE.                                                      MAIN   80
C                                                                      MAIN   90
      DOUBLE PRECISION BLOCK(13500)                                    MAIN  100
      DOUBLE PRECISION BETA, SUMLNX                                    MAIN  110
      LOGICAL ECHO, FAULT, GOF, HALT, NOGOOD                           MAIN  120
C                                                                      MAIN  130
      DATA IUNIT /5/                                                   MAIN  140
      DATA ECHO /.TRUE./                                               MAIN  150
      DATA NFAIL, NSYS /5000, 1000/                                    MAIN  160
C     TOTAL DATA STORAGE EQUALS 1.5 TIMES THE NUMBER OF FAILURES       MAIN  170
C     PLUS 6 TIMES THE NUMBER OF SYSTEMS.                              MAIN  180
      DATA NTOTAL /13500/                                              MAIN  190
C                                                                      MAIN  200
      WRITE (6,20)                                                     MAIN  210
C                                                                      MAIN  220
C     ALLOCATE STORAGE BASED ON MAXIMUM NUMBERS OF FAILURES            MAIN  230
C     AND SYSTEMS.                                                     MAIN  240
      NS2=NSYS*2                                                       MAIN  250
      I1=1                                                             MAIN  260
      I2=I1+NFAIL                                                      MAIN  270
      I3=I2+NS2                                                        MAIN  280
      I4=I3+NS2                                                        MAIN  290
      I5=I4+NS2                                                        MAIN  300
      ITOTAL=I5+(NFAIL/2)                                              MAIN  310
      IF (MOD(NFAIL,2).EQ.0) ITOTAL=ITOTAL-1                           MAIN  320
      IF (ITOTAL.LE.NTOTAL) GO TO 10                                   MAIN  330
      WRITE (6,30) ITOTAL,NTOTAL                                       MAIN  340
      STOP                                                             MAIN  350
C                                                                      MAIN  360
C                                                                      MAIN  370
C     BEGIN PROCESSING.                                                MAIN  380
   10 CALL INDATA (BLOCK(I1),BLOCK(I2),BLOCK(I5),NFAIL,NSYS,M,K,IUNIT, MAIN  390
     1ECHO,GOF,HALT,NOGOOD)                                            MAIN  400
      IF (HALT) STOP                                                   MAIN  410
      IF (NOGOOD) GO TO 10                                             MAIN  420
      CALL LOGS (BLOCK(I1),BLOCK(I2),BLOCK(I4),NFAIL,NSYS,M,K,SUMLNX)  MAIN  430
      CALL MLEW (SUMLNX,BLOCK(I2),BLOCK(I3),NFAIL,NSYS,M,K,BETA,       MAIN  440
     1FAULT)                                                           MAIN  450
      IF (.NOT.FAULT) CALL CVM (BLOCK(I1),BLOCK(I2),BLOCK(I4),BLOCK(I5),MAIN 460
     1NFAIL,NSYS,M,K,GOF,SUMLNX,BETA)                                  MAIN  470
      GO TO 10                                                         MAIN  480
C                                                                      MAIN  490
C                                                                      MAIN  500
   20 FORMAT (26H1 WEIBULL INTENSITY MODEL //1H0,30X,47H PARAMETER ESTIMMAIN 510
     1ATION AND GOODNESS OF FIT TEST //26H0 VERSION OF 12 NOV. 1979 ///)MAIN 520
   30 FORMAT (33H1 AMOUNT OF STORAGE REQUESTED IS ,I6,40H WORDS.  AMOUNTMAIN 530
     1 OF STORAGE AVAILABLE IS ,I5,25H WORDS.  PROGRAM ABORTED./)      MAIN  540
      END                                                              MAIN  550
      SUBROUTINE CVM (X,C,CLN,MARKER,NFAIL,NSYS,M,K,GOF,SUMLNX,BETA)    CVM   10
COMMENT     THIS SUBROUTINE PERFORMS THE CRAMER-VON MISES GOODNESS     CVM   20
C     OF FIT TEST.                                                     CVM   30
C                                                                      CVM   40
      DOUBLE PRECISION X(NFAIL), C(NSYS,2), CLN(NSYS,2), SUMLNX, BETA   CVM   50
      DOUBLE PRECISION CM2, DENOM, SUMSQS, TQ, TWOM, UNBETA            CVM   60
      DOUBLE PRECISION TERM, TERM1, TERM2                              CVM   70
      INTEGER MARKER(NFAIL)                                            CVM   80
      LOGICAL GOF                                                      CVM   90
```

21

```
C
      WRITE (6,60) BETA,M                                          CVM 100
      DENOM=0.0D0                                                  CVM 110
      IPT=                                                         CVM 120
      MSAVE=M                                                      CVM 130
C                                                                  CVM 140
      DO 20 J=1,K                                                  CVM 150
      CALL LOCIP (MARKER,NFAIL,MSAVE,J,IB,IE)                      CVM 160
      IF (IE.LT.IB) GO TO 20                                       CVM 170
      L=IE-IB+1                                                    CVM 180
C     SORT THE FAILURES FOR EACH SYSTEM INTO INCREASING ORDER.    CVM 190
      IF (L.GT.1) CALL SORTX (X(IB),L)                            CVM 200
C     CHECK FOR TIME TRUNCATED TESTING.                          CVM 210
      TQ=C(J,2)                                                    CVM 220
      IF (DABS(X(IE)-TQ).GT.1.0D-08) GO TO 10                     CVM 230
C     TIE LAST FAILURE IS NOT USED FOR FAILURE TRUNCATED SYSTEMS. CVM 240
      SUMLNX=SUMLNX-DLOG(X(IE))                                   CVM 250
      IE=IE-1                                                      CVM 260
      M=M-1                                                        CVM 270
      L=L-1                                                        CVM 280
   10 DENOM=DENOM+(DBLE(FLOAT(L))*CLN(J,2))                       CVM 290
C     TRANSFORM THE FAILURES.                                     CVM 300
      CALL TRANSF (X,NFAIL,TQ,IB,IE,IPT)                          CVM 310
   20 CONTINUE                                                     CVM 320
C                                                                  CVM 330
C     SORT THE TRANSFORMED FAILURES INTO INCREASING ORDER.        CVM 340
      CALL SORTX (X,M)                                             CVM 350
C     UNBIASED ESTIMATE OF BETA.                                  CVM 360
      DENOM=DENOM-SUMLNX                                           CVM 370
      UNBETA=DBLE(FLOAT(M-1))/DENOM                               CVM 380
      WRITE (6,90) UNBETA                                         CVM 390
C                                                                  CVM 400
      IF (GOF) GO TO 30                                            CVM 410
      WRITE (6,80)                                                CVM 420
      GO TO 50                                                     CVM 430
   30 SUMSQS=0.0D0                                                 CVM 440
      TWOM=DBLE(FLOAT(2*M))                                        CVM 450
      DO 40 I=1,M                                                  CVM 460
      TERM1=X(I)**UNBETA                                          CVM 470
      TERM2=DBLE(FLOAT(2*I-1))/TWOM                               CVM 480
      TERM=TERM1-TERM2                                             CVM 490
      SUMSQS=SUMSQS+TERM**2                                       CVM 500
   40 CONTINUE                                                     CVM 510
C     CRAMER-VON MISES STATISTIC.                                 CVM 520
      CMS=SUMSQS+(1.0D0/DBLE(FLOAT(12*M)))                        CVM 530
      WRITE (6,70) CM2,M                                          CVM 540
C                                                                  CVM 550
   50 RETURN                                                       CVM 560
C                                                                  CVM 570
   60 FORMAT (42H1 CRAMER - VON MISES GOODNESS OF FIT TEST.///18H0ESTIMACVM 580
     1TED BETA = ,1PD15.7/22H0NUMBER OF FAILURES = ,I6)          CVM 590
   70 FORMAT (32H0CRAMER - VON MISES STATISTIC = ,1PD15.7/52H0REJECT THECVM 600
     1 WEIBULL INTENSITY MODEL IF THE STATISTIC/48H EXCEEDS THE APPROPRICVM 610
     2ATE CRITICAL VALUE FOR M = ,I5/)                           CVM 620
   80 FORMAT (77H0THE CRAMER - VON MISES GOODNESS OF FIT TEST IS NOT APPCVM 630
     1ROPRIATE FOR THIS CASE/58H BECAUSE ONE OR MORE SYSTEMS HAVE NON-ZECVM 640
     2RO STARTING TIMES./)                                       CVM 650
   90 FORMAT (29H0UNBIASED ESTIMATE OF BETA = ,1PD15.7///)       CVM 660
      END                                                         CVM 670
      SUBROUTINE INDATA (X,C,MARKER,NFAIL,NSYS,M,K,IUNIT,ECHO,GOF,HALT, INDAA 10
     1 NGOOD)                                                     INDAA 20
COMMENT       THIS SUBROUTINE READS THE INPUT DATA ARRANGED AS    INDAA 30
C                                                                 INDAA 40
C     FLOWS:                                                      INDAA 50
```

22

```
C   BEGINNING AND ENDING TIMES FOR FIRST SYSTEM, FAILURE          INDAA 60
C     TIMES FOR FIRST SYSTEM, NEGATIVE VALUE AS TRAILER.          INDAA 70
C   BEGINNING AND ENDING TIMES FOR SECOND SYSTEM, FAILURE         INDAA 80
C     TIMES FOR SECOND SYSTEM, NEGATIVE VALUE FOR TRAILER.        INDAA 90
C     ...                                                         INDAA100
C     ...                                                         INDAA110
C   BEGINNING AND ENDING TIMES FOR K-TH SYSTEM, FAILURE           INDAA120
C     TIMES FOR K-TH SYSTEM, NEGATIVE VALUE AS TRAILER.           INDAA130
C     NEGATIVE VALUE TO MARK END OF CASE.                         INDAA140
C     (REPEAT ABOVE FOR AS MANY CASES AS NEEDED.)                 INDAA150
C     NEGATIVE VALUE TO MARK END OF RUN.                          INDAA160
C                                                                 INDAA170
C   INPUT IS FREE-FIELD, REQUIRING ONLY THAT AT LEAST ONE BLANK   INDAA180
C   COLUMN SEPARATE ADJACENT VALUES.                              INDAA190
C                                                                 INDAA200
C   FOR CASES WITH MORE THAN ONE SYSTEM, THE ROUTINE WILL         INDAA210
C   ACCEPT SOME (BUT NOT ALL) SYSTEMS WITH NO FAILURES.           INDAA220
C                                                                 INDAA230
      DOUBLE PRECISION X(NFAIL), C(NSYS,2)                        INDAA240
      INTEGER MARKER(NFAIL)                                       INDAA250
      LOGICAL ECHO, GOF, HALT, NOGOOD                             INDAA260
C                                                                 INDAA270
      GOF=.TRUE.                                                  INDAA280
      HALT=.FALSE.                                                INDAA290
      NOGOOD=.FALSE.                                              INDAA300
      WRITE (6,130)                                               INDAA310
C                                                                 INDAA320
      J=1                                                         INDAA330
      I=1                                                         INDAA340
C                                                                 INDAA350
C         BEGIN INPUT CYCLE.                                      INDAA360
C                                                                 INDAA370
C                                                                 INDAA380
   10 CALL INREAL (IUNIT,C(J,1))                                  INDAA390
C         NEGATIVE VALUE TO MARK THE END OF THIS CASE.            INDAA400
      IF (C(J,1).LT.0.0D0) GO TO 50                               INDAA410
      IF (DABS(C(J,1)).GT.1.0D-08) GOF=.FALSE.                    INDAA420
      CALL INREAL (IUNIT,C(J,2))                                  INDAA430
      J=J+1                                                       INDAA440
      IF (J.LE.NSYS) GO TO 20                                     INDAA450
      WRITE (6,100) J                                             INDAA460
      NOGOOD=.TRUE.                                               INDAA470
C                                                                 INDAA480
   20 CALL INREAL (IUNIT,X(I))                                    INDAA490
C         NEGATIVE VALUE TO MARK THE END OF THIS SYSTEM.          INDAA500
      IF (X(I).LT.0.0D0) GO TO 10                                 INDAA510
      IF (X(I).GT.1.0D-08) GO TO 30                               INDAA520
      WRITE (6,110)                                               INDAA530
      NOGOOD=.TRUE.                                               INDAA540
C                                                                 INDAA550
C         EACH FAILURE MUST FALL WITHIN THE TEST PERIOD.          INDAA560
   30 IF (C(J-1,1).LE.X(I).AND.X(I).LE.C(J-1,2)) GO TO 40         INDAA570
      WRITE (6,120) X(I),C(J-1,1),C(J-1,1),C(J-1,2)               INDAA580
      NOGOOD=.TRUE.                                               INDAA590
   40 MARKER(I)=J-1                                               INDAA600
      I=I+1                                                       INDAA610
      IF (I.LE.NFAIL) GO TO 20                                    INDAA620
      WRITE (6,90) I                                              INDAA630
      NOGOOD=.TRUE.                                               INDAA640
      GO TO 20                                                    INDAA650
C                                                                 INDAA660
C         END OF INPUT CYCLE.                                     INDAA670
C                                                                 INDAA680
   50 M=                                                          INDAA690
```

23

```
      IF (M.LE.0) GO TO 60                                               INDAA700
C                     TOTAL NUMBER OF FAILURES.                          INDAA710
      WRITE (6,70) M                                                     INDAA720
      K=J-1                                                              INDAA730
C                     TOTAL NUMBER OF SYSTEMS.                           INDAA740
      WRITE (6,80) K                                                     INDAA750
      IF (ECHO) CALL REPEAT (X,C,MARKER,NFAIL,NSYS,M,K)                  INDAA760
      RETURN                                                             INDAA770
C     ---------------------------------------------------------------   INDAA780
C                     END OF RUN.                                        INDAA790
C     ---------------------------------------------------------------   INDAA800
   60 HALT=.TRUE.                                                        INDAA810
      WRITE (6,140)                                                      INDAA820
      RETURN                                                             INDAA830
C                                                                        INDAA840
   70 FORMAT (28H0TOTAL NUMBER OF FAILURES = ,I5)                        INDAA850
   80 FORMAT (28H0TOTAL NUMBER OF SYSTEMS   = ,I5)                       INDAA860
   90 FORMAT (18H0 INDATA ERROR -- ,I6,22H IS TOO MANY FAILURES./)       INDAA870
  100 FORMAT (18H0 INDATA ERROR -- ,I6,21H IS TOO MANY SYSTEMS./)        INDAA880
  110 FORMAT (52H0 INDATA ERROR -- A FAILURE AT 0.0000000 WAS INPUT./71  INDAA890
     1H0 THE PROBABILITY OF SUCH A FAILURE TIME IS 0.0 ACCORDING TO THE  INDAA900
     2MODEL./)                                                           INDAA910
  120 FORMAT (33H0 INDATA ERROR -- THE FAILURE AT ,1PD10.3/44H DOES NOT  INDAA920
     1 FALL WITHIN THE TEST PERIOD FROM ,1PD10.3,4H TO ,1PD10.3/)        INDAA930
  130 FORMAT (19H1 DATA INPUT PHASE.//)                                  INDAA940
  140 FORMAT (28H0 PROGRAM RUN ENDS NORMALLY./1H1)                       INDAA950
      END                                                                INDAA960
      SUBROUTINE LOCIP (MARKER,NFAIL,M,J,IB,IE)                          LOCIP 10
COMMENT          THIS SUBROUTINE LOCATES THE FIRST AND THE LAST FAILURES LOCIP 20
C     ASSOCIATED WITH SYSTEM 'J'.  ON RETURN, 'IB' AND 'IE' INDEX        LOCIP 30
C     THESE FAILURES.                                                    LOCIP 40
C                                                                        LOCIP 50
      INTEGER MARKER(NFAIL)                                              LOCIP 60
C                                                                        LOCIP 70
      IB=0                                                               LOCIP 80
      IE=-1                                                              LOCIP 90
      DO 10 I=1,M                                                        LOCIP100
      IF (MARKER(I).LT.J) GO TO 10                                       LOCIP110
      IF (MARKER(I).GT.J) GO TO 20                                       LOCIP120
C                     EQUALITY.                                          LOCIP130
      IE=I                                                               LOCIP140
      IF (IB.EQ.0) IB=I                                                  LOCIP150
   10 CONTINUE                                                           LOCIP160
C                                                                        LOCIP170
   20 RETURN                                                             LOCIP180
      END                                                                LOCIP190
      SUBROUTINE LOGS (X,C,CLN,NFAIL,NSYS,M,K,SUMLNX)                    LOGS  10
COMMENT          THIS SUBROUTINE TAKES LOGARITHMS.                       LOGS  20
C                                                                        LOGS  30
      DOUBLE PRECISION X(NFAIL), C(NSYS,2), CLN(NSYS,2)                  LOGS  40
      DOUBLE PRECISION SUMLNX                                            LOGS  50
C                                                                        LOGS  60
C                     LOGARITHMS OF BEGINNING AND ENDING TIMES.          LOGS  70
      DO 20 I=1,K                                                        LOGS  80
      CLN(I,2)=DLOG(C(I,2))                                              LOGS  90
      IF (C(I,1).LE.0.0D0) GO TO 10                                      LOGS 100
      CLN(I,1)=DLOG(C(I,1))                                              LOGS 110
      GO TO 20                                                           LOGS 120
   10 C(I,1)=0.0D0                                                       LOGS 130
      CLN(I,1)=0.0D0                                                     LOGS 140
   20 CONTINUE                                                           LOGS 150
C                                                                        LOGS 160
C                     SUM OF LOGARITHMS OF FAILURES.                     LOGS 170
      SUMLNX=0.0D0                                                       LOGS 180
```

24

```
      DO 30 I=1,M                                                       LOGS 190
      SUMLNX=SUMLNX+DLOG(X(I))                                          LOGS 200
   30 CONTINUE                                                          LOGS 210
C                                                                       LOGS 220
      RETURN                                                            LOGS 230
      END                                                               LOGS 240
      SUBROUTINE MLEW (SUMLNX,C,CB,CLN,NSYS,M,K,BETA,NOGOOD)            MLEW  10
COMMENT    THIS SUBROUTINE CALCULATES THE MAXIMUM LIKELIHOOD           MLEW  20
C       ESTIMATES OF THE PARAMETERS BETA AND LAMBDA OF THE WEIBULL     MLEW  30
C       INTENSITY FUNCTION R(X)=LAMBDA*BETA*X**(BETA-1.0).             MLEW  40
C       BETA IS DETERMINED BY AN ITERATIVE PROCESS WHICH               MLEW  50
C       EXAMINES THE SIGNED DIFFERENCE OF A CONSTANT MINUS A FUNCTION  MLEW  60
C       OF BETA AS BETA IS DECREASED FROM A LARGE INITIAL ESTIMATE BY  MLEW  70
C       NON-POSITIVE POWERS OF 10.0 UNTIL AN EPSILON TOLERANCE IS      MLEW  80
C       SATISFIED OR BETA IS LESS THAN 0.00000001 IN VALUE.            MLEW  90
C       LAMBDA IS CALCULATED BASED ON BETA.                           MLEW 100
C                                                                       MLEW 110
      DOUBLE PRECISION C(NSYS,2), CB(NSYS,2), CLN(NSYS,2)              MLEW 120
      DOUBLE PRECISION BETA, LAMBDA, EPSILN, SUMLNX                    MLEW 130
      DOUBLE PRECISION DENOM, DENOM1, DENOM2, TOP, TOP1, TOP2          MLEW 140
      DOUBLE PRECISION A, ABDI, ADJ, D, DIFF, TOTFAL                   MLEW 150
      LOGICAL NOGOOD                                                    MLEW 160
C                                                                       MLEW 170
      WRITE (6,110)                                                     MLEW 180
      BETA=1.0D+01                                                      MLEW 190
      EPSILN=1.0D-05                                                    MLEW 200
      TOTFAL=DBLE(FLOAT(M))                                             MLEW 210
      NPDIFF=0                                                          MLEW 220
      NOGOOD=.FALSE.                                                    MLEW 230
C             CONSTANT NOT INVOLVING BETA.                             MLEW 240
      A=SUMLNX/TOTFAL                                                   MLEW 250
      WRITE (6,130) A                                                   MLEW 260
C                                                                       MLEW 270
C  ---------- BEGIN ITERATION PROCEDURE. ----------                    MLEW 280
C                                                                       MLEW 290
      ADJ=1.0D0                                                         MLEW 300
      IZERO=0                                                           MLEW 310
      WRITE (6,150)                                                     MLEW 320
   10 TOP1=0.0D0                                                        MLEW 330
      TOP2=0.0D0                                                        MLEW 340
      DENOM1=0.0D0                                                      MLEW 350
      DENOM2=0.0D0                                                      MLEW 360
      DO 20 I=1,K                                                       MLEW 370
      DO 20 J=1,2                                                       MLEW 380
      CB(I,J)=C(I,J)**BETA                                              MLEW 390
   20 CONTINUE                                                          MLEW 400
C                                                                       MLEW 410
      DO 30 I=1,K                                                       MLEW 420
      TOP1=TOP1+CB(I,1)*CLN(I,1)                                        MLEW 430
      TOP2=TOP2+CB(I,2)*CLN(I,2)                                        MLEW 440
      DENOM1=DENOM1+CB(I,1)                                             MLEW 450
      DENOM2=DENOM2+CB(I,2)                                             MLEW 460
   30 CONTINUE                                                          MLEW 470
C                                                                       MLEW 480
      TOP=TOP2-TOP1                                                     MLEW 490
      DENOM=DENOM2-DENOM1                                               MLEW 500
      D=(TOP/DENOM)-(1.0D0/BETA)                                        MLEW 510
      DIFF=A-D                                                          MLEW 520
      LAMBDA=TOTFAL/DENOM                                               MLEW 530
      WRITE (6,120) BETA,D,DIFF,LAMBDA                                  MLEW 540
      ABDI=DABS(DIFF)                                                   MLEW 550
      IF (ABDI.LE.EPSILN) GO TO 50                                      MLEW 580
```

25

```
      IF (DIFF.GT.0.0D0) GO TO 40                                       MLEW 590
      NPDIFF=0                                                          MLEW 600
      BETA=BETA-ADJ                                                     MLEW 610
      IF (BETA.LE.1.0D-15) GO TO 60                                     MLEW 620
      GO TO 10                                                          MLEW 630
C                                                                       MLEW 640
C        BETA TOO SMALL -- DECREASE STEP SIZE AND USE PREVIOUS BETA.    MLEW 650
C                                                                       MLEW 660
   40 BETA=BETA+ADJ                                                     MLEW 670
      NPDIFF=NPDIFF+1                                                   MLEW 680
      IF (NPDIFF.GT.10) GO TO 70                                        MLEW 690
      ADJ=1.0D-1*ADJ                                                    MLEW 700
      IF (DABS(DLOG10(DABS(BETA))-DLOG10(ADJ)).GT.1.5D+1) GO TO 80      MLEW 710
      BETA=BETA-ADJ                                                     MLEW 720
      GO TO 10                                                          MLEW 730
C                                                                       MLEW 740
C        EPSILON TOLERANCE MET.                                         MLEW 750
C                                                                       MLEW 760
   50 WRITE (6,140) BETA,LAMBDA,ABDI,EPSILN,ADJ                         MLEW 770
      GO TO 100                                                         MLEW 780
C                                                                       MLEW 790
   60 IZERO=IZERO+1                                                     MLEW 800
      IF (IZERO.LE.8) GO TO 40                                          MLEW 810
      WRITE (6,160)                                                     MLEW 820
      GO TO 90                                                          MLEW 830
   70 WRITE (6,170)                                                     MLEW 840
      GO TO 90                                                          MLEW 850
   80 WRITE (6,180)                                                     MLEW 860
   90 NOGOOD=.TRUE.                                                     MLEW 870
  100 RETURN                                                            MLEW 880
C                                                                       MLEW 890
C                                                                       MLEW 900
  110 FORMAT (65H1 ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY MLEW 920
     1 FUNCTION./)                                                      MLEW 930
  120 FORMAT (1H ,4(1PD16.9,4X))                                        MLEW 940
  130 FORMAT (35H0CONSTANT NOT INVOLVING BETA:    A = ,1PD15.7/)        MLEW 950
  140 FORMAT (30H1THE FINAL ESTIMATE OF BETA = ,1PD15.7//32H THE FINAL  MLEW 960
     1STIMATE OF LAMBDA = ,1PD15.7//16H CONVERGENCE TO ,1PD15.7/30H WHICH MLEW 970
     2H IS LESS THAN EPSILON = ,1PD15.7//24H THE FINAL STEP SIZE IS ,1PD MLEW 980
     315.7/)                                                           MLEW 990
  150 FORMAT (1H0//16H0 ESTIMATED BETA,5X,15H FUNCTION D(B'),8X,10H A - DMLEW1000
     1(B'),6X,17H ESTIMATED LAMBDA/)                                   MLEW1010
  160 FORMAT (41H0 MLEW ERROR -- BETA LESS THAN 0.00000001)            MLEW1020
  170 FORMAT (53H0 MLEW ERROR -- INITIAL ESTIMATE OF BETA IS TOO SMALL) MLEW1030
  180 FORMAT (50H0 MLEW ERROR -- STEP-SIZE HAS BECOME INSIGNIFICANT/26H MLEW1040
     1BETA NOT RESOLVABLE/)                                            MLEW1050
      END                                                              MLEW1060
      SUBROUTINE PRNTX (X,NFAIL,IB,IE)                                 PRNTX 10
COMMENT        THIS SUBROUTINE PRINTS THE FAILURES 'X(IB)' THROUGH     PRNTX 20
C        'X(IE)'.                                                      PRNTX 30
C                                                                      PRNTX 40
      DOUBLE PRECISION X(NFAIL)                                        PRNTX 50
      DOUBLE PRECISION XTEMP(5)                                        PRNTX 60
      DATA L /5/                                                       PRNTX 70
C                                                                      PRNTX 80
      IF (IE.LT.IB) GO TO 40                                           PRNTX 90
      IT=0                                                             PRNTX100
      DO 20 I=IB,IE                                                    PRNTX110
      IT=IT+1                                                          PRNTX120
      XTEMP(IT)=X(I)                                                   PRNTX130
      IF (I.EQ.IE) GO TO 10                                            PRNTX140
      IF (IT.LT.L) GO TO 20                                            PRNTX150
   10 WRITE (6,50) (XTEMP(J),J=1,IT)                                   PRNTX160
```

26

```
      IT=0
   20 CONTINUE
C
   30 RETURN
C
   40 WRITE (6,60)
      GO TO 30
C
   50 FORMAT (1H ,30X,1P5D10.3)                                        PRNTX170
   60 FORMAT (1H ,30X,12H NO FAILURES)                                 PRNTX180
      END                                                              PRNTX190
      SUBROUTINE REPEAT (X,C,MARKER,NFAIL,NSYS,M,K)                    PRNTX200
COMMENT      THIS SUBROUTINE REPEATS THE INPUT DATA.                   PRNTX210
C                                                                      PRNTX220
      DOUBLE PRECISION X(NFAIL), C(NSYS,2)                             PRNTX230
      INTEGER MARKER(NFAIL)                                            PRNTX240
C                                                                      PRNTX250
      WRITE (6,20)                                                     PRNTX260
C                                                                      PRNTX270
      DO 10 J=1,K                                                      REPET 10
      WRITE (6,30) C(J,1),C(J,2)                                       REPET 20
      CALL LOCIP (MARKER,NFAIL,M,J,IB,IE)                              REPET 30
      CALL PRNTX (X,NFAIL,IB,IE)                                       REPET 40
   10 CONTINUE                                                         REPET 50
C                                                                      REPET 60
      RETURN                                                           REPET 70
C                                                                      REPET 80
   20 FORMAT (1H0/34H0SYSTEM STARTING AND ENDING TIMES./1H0,35X,10H FAILREPET 90
     1URES./)                                                          REPET100
   30 FORMAT (1H ,1P2D12.3)                                            REPET110
      END                                                              REPET120
      SUBROUTINE SORTX (X,N)                                           REPET130
COMMENT      THIS SUBROUTINE SORTS THE VECTOR X INTO INCREASING ORDER. REPET140
C                                                                      REPET150
      DOUBLE PRECISION X(N)                                            REPET160
      DOUBLE PRECISION T                                               REPET170
      M=N                                                              REPET180
   10 M=M/2                                                            REPET190
      IF (M.EQ.0) RETURN                                               REPET200
      K=N-M+1                                                          SORTX 10
      J=1                                                              SORTX 20
   20 I=J                                                              SORTX 30
   30 L=I+M                                                            SORTX 40
      IF (X(I).GT.X(L)) GO TO 50                                       SORTX 50
   40 J=J+1                                                            SORTX 60
      IF (J-K) 20,10,10                                                SORTX 70
   50 T=X(L)                                                           SORTX 80
      X(L)=X(I)                                                        SORTX 90
      X(I)=T                                                           SORTX100
      I=I-M                                                            SORTX110
      IF (I) 40,40,30                                                  SORTX120
      END                                                              SORTX130
      SUBROUTINE TRANSF (X,NFAIL,TQ,IB,IE,IPT)                         SORTX140
COMMENT      THIS SUBROUTINE TRANSFORMS THE FAILURE TIMES.             SORTX150
C                                                                      SORTX160
      DOUBLE PRECISION X(NFAIL), TQ                                    SORTX170
C                                                                      SORTX180
      J=IPT                                                            SORTX190
C                                                                      SORTX200
      DO 10 I=IB,IE                                                    SORTX210
      X(J)=X(I)/TQ                                                     TRANF 10
      J=J+1                                                            TRANF 20
   10 CONTINUE                                                         TRANF 30
C                                                                      TRANF 40
                                                                       TRANF 50
                                                                       TRANF 60
                                                                       TRANF 70
                                                                       TRANF 80
                                                                       TRANF 90
                                                                       TRANF100
                                                                       TRANF110
                                                                       TRANF120
```

```
      IPT=J                                                       TRANF130
C                                                                 TRANF140
      RETURN                                                      TRANF150
      END                                                         TRANF160
      SUBROUTINE INREAL (CHANNL,DESTIN)                           INREL 10
COMMENT   FREE FIELD READ.                                        INREL 20
C         (A FORTRAN TRANSLATION OF ACM ALGORITHM 239.)           INREL 30
C                                                                 INREL 40
C         EACH CALL OF THIS SUBROUTINE WILL READ ONE REAL NUMBER  INREL 50
C         FROM UNIT 'CHANNL', CONVERT IT, AND STORE IT IN 'DESTIN'.INREL 60
C                                                                 INREL 70
      INTEGER CHANNL                                              INREL 80
      DOUBLE PRECISION DESTIN                                     INREL 90
      REAL SIG, FP, D                                             INREL100
      INTEGER ESIG, EP, IP, CH                                    INREL110
      INTEGER CHAR, UNSGND                                        INREL120
C                                                                 INREL130
      SIG=1.0                                                     INREL140
      EP=0                                                        INREL150
      FP=0.0                                                      INREL160
C                                                                 INREL170
   10 CH=CHAR(CHANNL)                                             INREL180
C         SUPPRESS INITIAL BLANKS.                                INREL190
      IF (CH.EQ.14) GO TO 10                                      INREL200
C         12 = '+', AND 11 = '-'.                                 INREL210
      IF (CH.NE.12) GO TO 20                                      INREL220
      CH=CHAR(CHANNL)                                             INREL230
      GO TO 30                                                    INREL240
   20 IF (CH.NE.11) GO TO 30                                      INREL250
      SIG=-1.0                                                    INREL260
      CH=CHAR(CHANNL)                                             INREL270
   30 CONTINUE                                                    INREL280
      IF (CH.GT.10) GO TO 70                                      INREL290
      IF (CH.EQ.10) GO TO 40                                      INREL300
      IP=UNSGND(CHANNL,CH)                                        INREL310
      GO TO 50                                                    INREL320
   40 CONTINUE                                                    INREL330
      IP=0                                                        INREL340
   50 CONTINUE                                                    INREL350
      IF (CH.NE.10) GO TO 100                                     INREL360
      CH=CHAR(CHANNL)                                             INREL370
      FP=0.0                                                      INREL380
      IF (CH.GE.10) GO TO 100                                     INREL390
      D=0.1                                                       INREL400
C                                                                 INREL410
   60 FP=FP+FLOAT(CH)*D                                           INREL420
      D=D*0.1                                                     INREL430
C                                                                 INREL440
      CH=CHAR(CHANNL)                                             INREL450
      IF (CH.LT.10) GO TO 60                                      INREL460
      GO TO 100                                                   INREL470
   70 CONTINUE                                                    INREL480
      IF (CH.NE.13) GO TO 80                                      INREL490
      IP=1                                                        INREL500
      GO TO 90                                                    INREL510
   80 CONTINUE                                                    INREL520
      WRITE (6,180)                                               INREL530
      STOP                                                        INREL540
   90 CONTINUE                                                    INREL550
  100 CONTINUE                                                    INREL560
C                                                                 INREL570
      IF (CH.NE.13) GO TO 160                                     INREL580
      CH=CHAR(CHANNL)                                             INREL590
      ESIG=1                                                      INREL600
```

28

```
      IF (CH.NE.12.AND.CH.NE.14) GO TO 110                              INREL610
      CH=CHAR(CHANNL)                                                   INREL620
      GO TO 130                                                         INREL630
  110 CONTINUE                                                          INREL640
      IF (CH.NE.11) GO TO 120      NEGATIVE EXPONENT.                    INREL650
C                                                                       INREL660
      ESIG=-1                                                           INREL670
      CH=CHAR(CHANNL)                                                   INREL680
  120 CONTINUE                                                          INREL690
  130 CONTINUE                                                          INREL700
      IF (CH.GE.10) GO TO 140                                           INREL710
      EP=UNSGND(CHANNL,CH)*ESIG                                         INREL720
      GO TO 150                                                         INREL730
  140 CONTINUE                                                          INREL740
      WRITE (6,190)                                                     INREL750
      STOP                                                              INREL760
  150 CONTINUE                                                          INREL770
  160 CONTINUE                                                          INREL780
C                                                                       INREL790
      IF (CH.NE.14) GO TO 170                                           INREL800
C                                                                       INREL810
      DESTIN=DBLE(SIG*(FLOAT(IP)+FP)*(10.0**EP))                        INREL820
C                                                                       INREL830
      RETURN                                                            INREL840
  170 WRITE (6,200)                                                     INREL850
      STOP                                                              INREL860
C                                                                       INREL870
  180 FORMAT (36H0 INREAL ERROR -- CH OUT OF RANGE.      /)             INREL880
  190 FORMAT (41H0 INREAL ERROR -- EXPONENT NOT DIGIT.   /)             INREL890
  200 FORMAT (53H0 INREAL ERROR -- NO BLANK FOUND BETWEEN DATA VALUES./) INREL900
      END                                                               INREL910
COMMENT     'CHAR' RETURNS AN INTEGER VALUE FOR THE NEXT CHARACTER      CHAR  10
C           ON UNIT 'CHANNL'.                                           CHAR  20
      INTEGER FUNCTION CHAR(CHANNL)                                     CHAR  30
C                                                                       CHAR  40
      INTEGER C                                                         CHAR  50
      INTEGER CHANNL                                                    CHAR  60
      INTEGER STRING(15)                                                CHAR  70
      DATA STRING(1), STRING(2), STRING(3), STRING(4), STRING(5), STRING CHAR  80
     1(6), STRING(7), STRING(8), STRING(9), STRING(10), STRING(11), STRI CHAR  82
     2NG(12), STRING(13), STRING(14), STRING(15) /1H0, 1H1, 1H2, 1H3, 1H CHAR  84
     3, 1H5, 1H6, 1H7, 1H8, 1H9, 1H., 1H-, 1H+, 1HE, 1H /                CHAR  90
      DATA LSTR /15/                                                    CHAR 100
      CALL INSYML (CHANNL,STRING,LSTR,C)                                CHAR 110
C             IS CHARACTER LEGAL?                                       CHAR 120
      IF (C.LE.0) GO TO 10                                              CHAR 130
      CHAR=C-1                                                          CHAR 140
      RETURN                                                            CHAR 150
   10 WRITE (6,20)                                                      CHAR 160
      STOP                                                              CHAR 170
C                                                                       CHAR 180
   20 FORMAT (58H0 CHAR ERROR -- ILLEGAL INPUT CHARACTER.  PROGRAM ABORT CHAR 190
     1ED./)                                                             CHAR 192
      END                                                               CHAR 200
      SUBROUTINE INSYML (IUNIT,STRING,LSTR,I)                           INSYL 10
COMMENT     THIS SUBROUTINE EXAMINES THE NEXT CHARACTER ON 'IUNIT'      INSYL 20
C           AND DETERMINES ITS POSITION NUMBER 'I' WITHIN THE 'STRING' OF INSYL 30
C           LENGTH 'LSTR'.                                              INSYL 40
C                                                                       INSYL 50
      INTEGER STRING(LSTR)                                              INSYL 60
      INTEGER BUFFER(160)                                               INSYL 70
C                                                                       INSYL 80
      DATA IP /0/                                                       INSYL 90
      DATA LENGTH /80/                                                  INSYL100
```

29

```
C
C      IF (IP.NE.0) GO TO 10                                           INSYL110
C            FILL INPUT BUFFER.                                        INSYL120
       READ (IUNIT,20) (BUFFER(J),J=1,LENGTH)                          INSYL130
   10 CONTINUE                                                         INSYL140
C                                                                      INSYL150
       IP=IP+1                                                         INSYL160
       IC=IP                                                           INSYL170
C                                                                      INSYL180
C         MATCH THE CHARACTER.                                         INSYL190
       CALL MATCH (BUFFER(IC),STRING,LSTR,I)                           INSYL200
C                                                                      INSYL210
C         IF POINTER 'IP' HAS REACHED THE END OF A LINE, RESET IT.     INSYL220
       IF (IP.EQ.LENGTH) IP=0                                          INSYL230
C                                                                      INSYL240
       RETURN                                                          INSYL250
C                                                                      INSYL260
   20 FORMAT (128A1)                                                   INSYL270
       END                                                             INSYL280
       SUBROUTINE MATCH (CHAR,STRING,LSTR,IP)                          INSYL290
COMMENT      THIS SUBROUTINE FINDS THE POSITION 'IP' OF 'CHAR' IN      INSYL300
C      'STRING' WHICH HAS A LENGTH OF 'LSTR'.                          INSYL310
C                                                                      MATCH 10
       INTEGER CHAR, STRING(LSTR)                                      MATCH 20
       IP=0                                                            MATCH 30
       DO 10 I=1,LSTR                                                  MATCH 40
       IF (STRING(I).NE.CHAR) GO TO 10                                 MATCH 50
       IP=I                                                            MATCH 60
       GO TO 20                                                        MATCH 70
   10 CONTINUE                                                         MATCH 80
       WRITE (6,30) CHAR                                               MATCH 90
   20 RETURN                                                           MATCH100
C                                                                      MATCH110
   30 FORMAT (31H0 MATCH ERROR -- THE CHARACTER ,A1,16H IS NOT MATCHED./MATCH120
      1)                                                               MATCH130
       END                                                             MATCH140
       INTEGER FUNCTION UNSGND(CHANNL,CH)                              MATCH150
COMMENT      THIS FUNCTION RETURNS THE NEXT UNSIGNED INTEGER FROM      MATCH160
C      'CHANNL'.                                                       MATCH170
C                                                                      UNSG 10
       INTEGER CHANNL, CH                                              UNSG 20
       INTEGER CHAR                                                    UNSG 30
       INTEGER U                                                       UNSG 40
       U=0                                                             UNSG 50
   10 U=10*U+CH                                                        UNSG 60
       CH=CHAR(CHANNL)                                                 UNSG 70
       IF (CH.LT.10) GO TO 10                                          UNSG 80
       UNSGND=U                                                        UNSG 90
       RETURN                                                          UNSG100
       END                                                             UNSG110
                                                                       UNSG120
                                                                       UNSG130
                                                                       UNSG140
```

30

APPENDIX B

INPUT FOR TEST CASES

Next page is blank.

INPUT DATA

```
    0.0   200.0   4.3   4.4   10.2   23.5   23.8   26.4   74.0   77.1   92.1   197.2
-999.
    0.0   200.0   0.1   5.6   18.6   19.5   24.2   26.7   45.1   45.8   75.7   79.7   98.6
              120.1   161.8   180.6   190.8
    -999
    0.0   200.0   8.4   32.5   44.7   48.4   50.6   73.6   98.7   112.2   129.8   136.0
              195.8
    -999
    -999
    0.0   197.2   4.3   4.4   10.2   23.5   23.8   26.4   74.0   197.2   77.1   92.1
-999.
    0.0   190.8   0.1   5.6   18.6   19.5   24.2   26.7   45.1   45.8   75.7   79.7   98.6
              120.1   161.8   180.6   190.8
    -999
    0.0   195.8   195.8
                   8.4   32.5   44.7   48.4   50.6   73.6   98.7   112.2   129.8   136.0
    -999
    -999
 0  10  -99   15  -99   0  2.   4  12  17  -99   0  20  -99   0  10  -99
-99
    1.0   100.0   32.   44.   56.   75.   95.   -99.   -99.
12.2   45.3   10.0   14.5   16.8   457.9   -99  -99
    0.0   3256.3   .7   3.7   13.2   17.6   54.5   99.2   112.2   120.9   151.0   163.0
174.5   191.6   282.8
355.2   486.3   490.5   513.3   558.4   678.1   688.0   785.9   887.0   1010.7
 1029.1   1134.4   1136.1   1178.9   1259.7   1297.9   1419.7   1571.7
1629.8   1702.3   1928.9   2072.3   2525.2   2928.5   3016.4   3181.0
 3256.3
 -99.   -99.
    0.0   79.0   0.0   20   45.   62   64   79   68   -8  -7
    456.   789.   456   456.1   457   458   489   -99   -99
0.0   800.0   45.   456.   467.   477.   484.   492.
0.12   564.0   65.0   78.   89.   99.1   -99  -99
 0.  10.  2.  3.  4.  5.  6.   8.  -99.   0.  30.  -99.   0.  50.  41.  42.  43.  44.  -99
    0.  60.  51.  52.  53.  -99.  -99.
    -99
```

Next page is blank.

33

APPENDIX C

OUTPUT FOR TEST CASES

WEIBULL INTENSITY MODEL

PARAMETER ESTIMATION AND GOODNESS OF FIT TEST

VERSION OF 12 NOV. 1979

37

DATA INPUT PHASE.

TOTAL NUMBER OF FAILURES =   36

TOTAL NUMBER OF SYSTEMS =   3

SYSTEM STARTING AND ENDING TIMES.

FAILURES.

0.	2.000D+02					
		4.300D+00	4.400D+00	1.020D+01	2.350D+01	2.380D+01
		2.640D+01	7.400D+01	7.710D+01	9.210D+01	1.972D+02
0.	2.000D+02					
		1.000D-01	5.600D+00	1.860D+01	1.950D+01	2.420D+01
		2.670D+01	4.510D+01	4.580D+01	7.570D+01	7.970D+01
		9.860D+01	1.201D+02	1.618D+02	1.606D+02	1.908D+02
0.	2.000D+02					
		8.400D+00	3.250D+01	4.470D+01	4.840D+01	5.060D+01
		7.360D+01	9.870D+01	1.122D+02	1.298D+02	1.360D+02
		1.958D+02				

ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY FUNCTION.

CONSTANT NOT INVOLVING BETA: A = 3.67318960+00

ESTIMATED BETA	FUNCTION D(8')	A - D(8')	ESTIMATED LAMBDA
1.000000000D+01	5.198317367D+00	-1.525127718D+00	1.171875000D-22
9.000000000D+00	5.187206255D+00	-1.514016607D+00	2.343750000D-20
8.000000000D+00	5.173317367D+00	-1.500127718D+00	4.687500000D-18
7.000000000D+00	5.155460224D+00	-1.482270575D+00	9.375000000D-16
6.000000000D+00	5.131650700D+00	-1.458461051D+00	1.875000000D-13
5.000000000D+00	5.098317367D+00	-1.425127718D+00	3.750000000D-11
4.000000000D+00	5.048317367D+00	-1.375127718D+00	7.500000000D-09
3.000000000D+00	4.964984033D+00	-1.291794384D+00	1.500000000D-06
2.000000000D+00	4.798317367D+00	-1.125127718D+00	3.000000000D-04
1.000000000D+00	4.298317367D+00	-6.251277178D-01	6.000000000D-02
9.000000000D-01	4.187206255D+00	-5.140166067D-01	1.019187879D-01
8.000000000D-01	4.048317367D+00	-3.751277178D-01	1.731239887D-01
7.000000000D-01	3.869745938D+00	-1.965562892D-01	2.940764514D-01
6.000000000D-01	3.631650700D+00	4.153894890D-02	4.995319244D-01
6.900000000D-01	3.849042004D+00	-1.758523554D-01	3.100777107D-01
6.800000000D-01	3.827729131D+00	-1.545394825D-01	3.269496291D-01
6.700000000D-01	3.805780053D+00	-1.325904043D-01	3.447395806D-01
6.600000000D-01	3.783165851D+00	-1.099762026D-01	3.634975173D-01
6.500000000D-01	3.759855828D+00	-8.666617931D-02	3.832761089D-01
6.400000000D-01	3.735817367D+00	-6.262771777D-02	4.041308914D-01
6.300000000D-01	3.711015779D+00	-3.782613047D-02	4.261204223D-01
6.200000000D-01	3.685414141D+00	-1.222449196D-02	4.493064454D-01
6.100000000D-01	3.658973104D+00	1.421654453D-02	4.737540642D-01
6.190000000D-01	3.682808481D+00	-9.618832469D-03	4.516933312D-01
6.180000000D-01	3.680194389D+00	-7.004740422D-03	4.540928971D-01
6.170000000D-01	3.677571824D+00	-4.382174819D-03	4.565052103D-01
6.160000000D-01	3.674940743D+00	-1.751094392D-03	4.589303387D-01
6.150000000D-01	3.672301106D+00	8.885423943D-04	4.613683502D-01
6.159000000D-01	3.674677165D+00	-1.487516437D-03	4.591735590D-01
6.158000000D-01	3.674413502D+00	-1.223852877D-03	4.594169082D-01
6.157000000D-01	3.674149752D+00	-9.601036705D-04	4.596603863D-01
6.156000000D-01	3.673885918D+00	-6.962687754D-04	4.599039935D-01
6.155000000D-01	3.673621997D+00	-4.323481501D-04	4.601477298D-01
6.154000000D-01	3.673357991D+00	-1.683417527D-04	4.603915953D-01
6.153000000D-01	3.673093898D+00	9.575045862D-05	4.606355900D-01
6.153900000D-01	3.673331585D+00	-1.419363938D-04	4.604159889D-01
6.153800000D-01	3.673305179D+00	-1.155301769D-04	4.604403839D-01
6.153700000D-01	3.673278772D+00	-8.912310164D-05	4.604647801D-01
6.153600000D-01	3.673252364D+00	-6.271516816D-05	4.604891776D-01
6.153500000D-01	3.673225955D+00	-3.630637637D-05	4.605135765D-01
6.153400000D-01	3.673199546D+00	-9.896726236D-06	4.605379766D-01

THE FINAL ESTIMATE OF BETA = 6.1534000D-01

THE FINAL ESTIMATE OF LAMBDA = 4.6053798D-01

CONVERGENCE TO 9.8967262D-06
WHICH IS LESS THAN EPSILON = 1.0000000D-05

THE FINAL STEP SIZE IS 1.0000000D-05

CRAMER — VON MISES GOODNESS OF FIT TEST.

ESTIMATED BETA = 6.1534000D-01

NUMBER OF FAILURES = 36

UNBIASED ESTIMATE OF BETA = 5.9824358D-01

CRAMER — VON MISES STATISTIC = 6.9530891D-02

REJECT THE WEIBULL INTENSITY MODEL IF THE STATISTIC
EXCEEDS THE APPROPRIATE CRITICAL VALUE FOR M = 36

DATA INPUT PHASE.

TOTAL NUMBER OF FAILURES =   36

TOTAL NUMBER OF SYSTEMS =   3

SYSTEM STARTING AND ENDING TIMES.

                                        FAILURES.

0.   1.972D+02
                            4.300D+00 4.400D+00 1.020D+01 2.350D+01 2.380D+01
                            2.640D+01 7.400D+01 1.972D+02 7.710D+01 9.210D+01
0.   1.908D+02
                            1.000D-01 5.600D+00 1.860D+01 1.950D+01 2.420D+01
                            2.670D+01 4.510D+01 4.580D+01 7.570D+01 7.970D+01
                            9.860D+01 1.201D+02 1.618D+02 1.806D+02 1.908D+02
0.   1.958D+02
                            1.958D+02 8.400D+00 3.250D+01 4.470D+01 4.840D+01
                            5.060D+01 7.360D+01 9.870D+01 1.122D+02 1.298D+02
                            1.360D+02

ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY FUNCTION.

CONSTANT NOT INVOLVING BETA: A =   3.6731896D+00

ESTIMATED BETA	FUNCTION D(B')	A - D(B')	ESTIMATED LAMBDA
1.0000000000D+01	5.172764354D+00	-1.499574705D+00	1.527402235D-22
9.0000000000D+00	5.161470246D+00	-1.488280598D+00	2.977461975D-20
8.0000000000D+00	5.147396282D+00	-1.474206633D+00	5.803087229D-18
7.0000000000D+00	5.129352022D+00	-1.456162373D+00	1.130813934D-15
6.0000000000D+00	5.105353381D+00	-1.432163732D+00	2.203137001D-13
5.0000000000D+00	5.071828974D+00	-1.398639326D+00	4.291501795D-11
4.0000000000D+00	5.021635991D+00	-1.348446342D+00	8.357833094D-09
3.0000000000D+00	4.938107813D+00	-1.264918165D+00	1.627398353D-06
2.0000000000D+00	4.771244493D+00	-1.098054844D+00	3.168174072D-04
1.0000000000D+00	4.271046084D+00	-5.978564354D-01	6.166495375D-02
9.0000000000D-01	4.159915038D+00	-4.867253891D-01	1.044615824D-01
8.0000000000D-01	4.021006197D+00	-3.478165480D-01	1.769595169D-01
7.0000000000D-01	3.842414799D+00	-1.692251504D-01	2.997715275D-01
6.0000000000D-01	3.604299575D+00	6.889007354D-02	5.078552206D-01
6.9000000000D-01	3.821708868D+00	-1.485192188D-01	3.159962865D-01
6.8000000000D-01	3.800393997D+00	-1.272043479D-01	3.330991836D-01
6.7000000000D-01	3.778442920D+00	-1.052532716D-01	3.511277462D-01
6.6000000000D-01	3.755826720D+00	-8.263707150D-02	3.701320741D-01
6.5000000000D-01	3.732514698D+00	-5.932504969D-02	3.901649784D-01
6.4000000000D-01	3.708474238D+00	-3.528458948D-02	4.112821287D-01
6.3000000000D-01	3.683670652D+00	-1.048100334D-02	4.335422075D-01
6.2000000000D-01	3.658067015D+00	1.512263417D-02	4.570707733D-01
6.2900000000D-01	3.681146920D+00	-7.957271513D-03	4.358334296D-01
6.2800000000D-01	3.678615152D+00	-5.425502957D-03	4.381367604D-01
6.2700000000D-01	3.676075308D+00	-2.885659220D-03	4.404522640D-01
6.2600000000D-01	3.673527350D+00	-3.377016037D-04	4.427800047D-01
6.2500000000D-01	3.670971240D+00	2.218408839D-03	4.451200472D-01
6.2590000000D-01	3.673272107D+00	-8.245808135D-05	4.430134543D-01
6.2580000000D-01	3.673016782D+00	1.728670081D-04	4.432470270D-01
6.2589000000D-01	3.673246578D+00	-5.692924357D-05	4.430368061D-01
6.2588000000D-01	3.673221048D+00	-3.139959008D-05	4.430601590D-01
6.2587000000D-01	3.673195518D+00	-5.869120840D-06	4.430835132D-01

43

THE FINAL ESTIMATE OF BETA = 6.2587000D-01

THE FINAL ESTIMATE OF LAMBDA = 4.4308351D-01

CONVERGENCE TO 5.8691208D-06
WHICH IS LESS THAN EPSILON = 1.0000000D-05

THE FINAL STEP SIZE IS 1.0000000D-05

CRAMER – VON MISES GOODNESS OF FIT TEST.

ESTIMATED BETA = 6.2587000D-01

NUMBER OF FAILURES = 36

UNBIASED ESTIMATE OF BETA = 5.5726048D-01

CRAMER – VON MISES STATISTIC = 1.3643898D-01

REJECT THE WEIBULL INTENSITY MODEL IF THE STATISTIC
EXCEEDS THE APPROPRIATE CRITICAL VALUE FOR M = 33

DATA INPUT PHASE.

TOTAL NUMBER OF FAILURES =   3

TOTAL NUMBER OF SYSTEMS =   5

SYSTEM STARTING AND ENDING TIMES.

FAILURES.

0.	1.000D+01

NO FAILURES

0.	1.500D+01

NO FAILURES

0.	2.000D+01

4.000D+00 1.200D+01 1.700D+01

0.	2.000D+01

NO FAILURES

0.	1.000D+01

NO FAILURES

ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY FUNCTION.

CONSTANT NOT INVOLVING BETA: A =   2.2348048D+00

ESTIMATED BETA	FUNCTION D(B')	A - D(B')	ESTIMATED LAMBDA
1.000000000D+01	2.887203645D+00	-6.523988600D-01	1.423376079D-13
9.000000000D+00	2.872928892D+00	-6.381241070D-01	2.818374480D-12
8.000000000D+00	2.854500246D+00	-6.196954611D-01	5.559376018D-11
7.000000000D+00	2.829967336D+00	-5.951625507D-01	1.090568288D-09
6.000000000D+00	2.796084828D+00	-5.612800428D-01	2.121781412D-08
5.000000000D+00	2.747210761D+00	-5.124059755D-01	4.076433121D-07
4.000000000D+00	2.672959541D+00	-4.381547563D-01	7.680000000D-06
3.000000000D+00	2.552119579D+00	-3.173147936D-01	1.403508772D-04
2.000000000D+00	2.329725823D+00	-9.492103762D-02	2.448979592D-03
1.000000000D+00	1.753356611D+00	4.814481741D-01	4.000000000D-02
1.900000000D+00	2.296695350D+00	-6.190056461D-02	3.248879162D-03
1.800000000D+00	2.260528724D+00	-2.572393895D-02	4.307107646D-03
1.700000000D+00	2.220711095D+00	1.409368974D-02	5.706009308D-03
1.790000000D+00	2.256720743D+00	-2.191595789D-02	4.430107818D-03
1.780000000D+00	2.252875780D+00	-1.807099482D-02	4.556588426D-03
1.770000000D+00	2.248993245D+00	-1.418846036D-02	4.686646926D-03
1.760000000D+00	2.245072537D+00	-1.026775178D-02	4.820383476D-03
1.750000000D+00	2.241113038D+00	-6.308252638D-03	4.957901000D-03
1.740000000D+00	2.237114117D+00	-2.309332347D-03	5.099305272D-03
1.730000000D+00	2.233075131D+00	1.729654209D-03	5.244704988D-03
1.739000000D+00	2.236712032D+00	-1.907247446D-03	5.113663630D-03
1.738000000D+00	2.236309546D+00	-1.504761241D-03	5.128062049D-03
1.737000000D+00	2.235906658D+00	-1.101873078D-03	5.142500642D-03
1.736000000D+00	2.235503367D+00	-6.985823011D-04	5.156979518D-03
1.735000000D+00	2.235099673D+00	-2.948882514D-04	5.171498790D-03
1.734000000D+00	2.234695575D+00	1.092097299D-04	5.186058568D-03
1.734900000D+00	2.235059282D+00	-2.544966411D-04	5.172952943D-03
1.734800000D+00	2.235018886D+00	-2.141009908D-04	5.174407501D-03
1.734700000D+00	2.234978486D+00	-1.737012999D-04	5.175862465D-03
1.734600000D+00	2.234938083D+00	-1.332975676D-04	5.177317834D-03
1.734500000D+00	2.234897675D+00	-9.288979346D-05	5.178773609D-03
1.734400000D+00	2.234857263D+00	-5.247797667D-05	5.180229789D-03
1.734300000D+00	2.234816847D+00	-1.206211661D-05	5.181686375D-03
1.734200000D+00	2.234776427D+00	2.835778740D-05	5.183143366D-03
1.734290000D+00	2.234812805D+00	-8.020308194D-06	5.181832056D-03

THE FINAL ESTIMATE OF BETA =   1.7342900D+00

THE FINAL ESTIMATE OF LAMBDA =   5.1818321D-03

CONVERGENCE TO   8.0203082D-06
WHICH IS LESS THAN EPSILON =   1.0000000D-05

THE FINAL STEP SIZE IS   1.0000000D-05

CRAMER – VON MISES GOODNESS OF FIT TEST.

ESTIMATED BETA =   1.7342900D+00

NUMBER OF FAILURES =   3

UNBIASED ESTIMATE OF BETA =   8.7612378D-01

CRAMER – VON MISES STATISTIC =   5.4305758D-02

REJECT THE WEIBULL INTENSITY MODEL IF THE STATISTIC
EXCEEDS THE APPROPRIATE CRITICAL VALUE FOR M =   3

DATA INPUT PHASE.

TOTAL NUMBER OF FAILURES =   5
TOTAL NUMBER OF SYSTEMS =   1

SYSTEM STARTING AND ENDING TIMES.

                              FAILURES.

   1.000D+00   1.000D+02

                    3.200D+01 4.400D+01 5.600D+01 7.500D+01 9.500D+01

ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY FUNCTION.

CONSTANT NOT INVOLVING BETA: A =   4.0293284D+00

ESTIMATED BETA	FUNCTION D(B')	A - D(B')	ESTIMATED LAMBDA
1.0000000000D+01	4.5051701860D+00	-4.758417395D-01	5.000000000D-20
9.0000000000D+00	4.494059075D+00	-4.647306284D-01	5.000000000D-18
8.0000000000D+00	4.480170186D+00	-4.508417395D-01	5.0000000000D-16
7.0000000000D+00	4.462313043D+00	-4.329845966D-01	5.000000000D-14
6.0000000000D+00	4.438503519D+00	-4.091750728D-01	5.0000000000D-12
5.0000000000D+00	4.4051701860D+00	-3.758417399D-01	5.000000001D-10
4.0000000000D+00	4.355170232D+00	-3.258417855D-01	5.000000050D-08
3.0000000000D+00	4.2718414580D+00	-2.425130113D-01	5.000005000D-06
2.0000000000D+00	4.105630749D+00	-7.630230255D-02	5.000500050D-04
1.0000000000D+00	3.6516870570D+00	3.776413900D-01	5.050505051D-02
1.9000000000D+00	4.079584382D+00	-5.025593598D-02	7.925722105D-04
1.8000000000D+00	4.050771688D+00	-2.144324100D-02	1.256258774D-03
1.7000000000D+00	4.018768973D+00	1.055947321D-02	1.991328615D-03
1.7900000000D+00	4.047722570D+00	-1.839412345D-02	1.315480002D-03
1.7800000000D+00	4.044641152D+00	-1.531270529D-02	1.377493745D-03
1.7700000000D+00	4.041526963D+00	-1.219851680D-02	1.442431753D-03
1.7600000000D+00	4.038379527D+00	-9.051080574D-03	1.510432003D-03
1.7500000000D+00	4.035198358D+00	-5.869911384D-03	1.581638988D-03
1.7400000000D+00	4.031982963D+00	-2.654516074D-03	1.656204028D-03
1.7300000000D+00	4.028732840D+00	5.956065495D-04	1.734285593D-03
1.7390000000D+00	4.0316595210D+00	-2.331074881D-03	1.663851262D-03
1.7380000000D+00	4.031335732D+00	-2.007285916D-03	1.671533818D-03
1.7370000000D+00	4.031011595D+00	-1.683148672C-03	1.679251859D-03
1.7360000000D+00	4.030687109D+00	-1.358662642D-03	1.687005549D-03
1.7350000000D+00	4.030362274D+00	-1.033827318D-03	1.694795052D-03
1.7340000000D+00	4.030037089D+00	-7.086421920D-04	1.702620534D-03
1.7330000000D+00	4.029711553D+00	-3.831067529D-04	1.710482162D-03
1.7320000000D+00	4.029385667D+00	-5.722049074D-05	1.718380103D-03
1.7310000000D+00	4.029059429D+00	2.690171060D-04	1.726314523D-03
1.7319000000D+00	4.029353059D+00	-2.461254957D-05	1.719171900D-03
1.7318000000D+00	4.029320448D+00	7.998905460D-06	1.719964063D-03

51

THE FINAL ESTIMATE OF BETA =   1.7318000D+00

THE FINAL ESTIMATE OF LAMBDA =   1.7199641D-03

CONVERGENCE TO   7.9989055D-06
WHICH IS LESS THAN EPSILON =   1.0000000D-05

THE FINAL STEP SIZE IS   1.0000000D-04

CRAMER - VON MISES GOODNESS OF FIT TEST.

ESTIMATED BETA =   1.7318000D+00

NUMBER OF FAILURES =   5

UNBIASED ESTIMATE OF BETA =   1.3892706D+00

THE CRAMER - VON MISES GOODNESS OF FIT TEST IS NOT APPROPRIATE FOR THIS CASE
BECAUSE ONE OR MORE SYSTEMS HAVE NON-ZERO STARTING TIMES.

DATA INPUT PHASE.

INDATA ERROR -- THE FAILURE AT   1.000D+01
DOES NOT FALL WITHIN THE TEST PERIOD FROM   1.220D+01 TO   4.530D+01

INDATA ERROR -- THE FAILURE AT   4.579D+02
DOES NOT FALL WITHIN THE TEST PERIOD FROM   1.220D+01 TO   4.530D+01

TOTAL NUMBER OF FAILURES =   4

TOTAL NUMBER OF SYSTEMS =   1

SYSTEM STARTING AND ENDING TIMES.

                            FAILURES.

   1.220D+01   4.530D+01
                      1.000D+01 1.450D+01 1.680D+01 4.579D+02

54

DATA INPUT PHASE.

TOTAL NUMBER OF FAILURES =   40

TOTAL NUMBER OF SYSTEMS =   1

SYSTEM STARTING AND ENDING TIMES.

                                        FAILURES.

      0.   3.256D+03

                              7.000D-01 3.700D+00 1.320D+01 1.760D+01 5.450D+01
                              9.920D+01 1.122D+02 1.209D+02 1.510D+02 1.630D+02
                              1.745D+02 1.916D+02 2.828D+02 3.552D+02 4.863D+02
                              4.905D+02 5.133D+02 5.584D+02 6.781D+02 6.880D+02
                              7.859D+02 8.870D+02 1.011D+03 1.029D+03 1.034D+03
                              1.136D+03 1.179D+03 1.260D+03 1.298D+03 1.420D+03
                              1.572D+03 1.630D+03 1.702D+03 1.929D+03 2.072D+03
                              2.525D+03 2.929D+03 3.016D+03 3.181D+03 3.256D+03

ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY FUNCTION.

CONSTANT NOT INVOLVING BETA: A =   6.0464988D+00

ESTIMATED BETA	FUNCTION D(B')	A - D(B')	ESTIMATED LAMBDA
1.000000000D+01	7.988346860D+00	-1.941848101D+00	2.984117360D-34
9.000000000D+00	7.977235749D+00	-1.930736990D+00	9.717181358D-31
8.000000000D+00	7.963346860D+00	-1.916848101D+00	3.164205766D-27
7.000000000D+00	7.945489718D+00	-1.898990958D+00	1.030360323D-23
6.000000000D+00	7.921680194D+00	-1.875181434D+00	3.351623321D-20
5.000000000D+00	7.888346860D+00	-1.841848101D+00	1.092541507D-16
4.000000000D+00	7.838346860D+00	-1.791848101D+00	3.557642908D-13
3.000000000D+00	7.755013527D+00	-1.708514768D+00	1.158475260D-09
2.000000000D+00	7.588346860D+00	-1.541848101D+00	3.772342990D-06
1.000000000D+00	7.088346860D+00	-1.041848101D+00	1.228388048D-02
9.000000000D-01	6.977235749D+00	-9.307369898D-01	2.758087392D-02
8.000000000D-01	6.838346860D+00	-7.918481009D-01	6.192706024D-02
7.000000000D-01	6.659775432D+00	-6.132766723D-01	1.390442087D-01
6.000000000D-01	6.421680194D+00	-3.751814342D-01	3.121945705D-01
5.000000000D-01	6.088346860D+00	-4.184810091D-02	7.009673453D-01
4.000000000D-01	5.588346860D+00	4.581518991D-01	1.573874966D+00
4.900000000D-01	6.047530534D+00	-1.031774382D-03	7.600200173D-01
4.800000000D-01	6.005013527D+00	4.148523242D-02	8.240475545D-01
4.890000000D-01	6.043357085D+00	3.141674139D-03	7.661922507D-01
4.899000000D-01	6.047113956D+00	-6.151962382D-04	7.606349965D-01
4.898000000D-01	6.046697208D+00	-1.984479930D-04	7.612504734D-01
4.897000000D-01	6.046280289D+00	2.184704576D-04	7.618664482D-01
4.897900000D-01	6.046655523D+00	-1.567638089D-04	7.613120484D-01
4.897800000D-01	6.046613837D+00	-1.150779227D-04	7.613736285D-01
4.897700000D-01	6.046572150D+00	-7.339033417D-05	7.614352135D-01
4.897600000D-01	6.046530461D+00	-3.170104328D-05	7.614968035D-01
4.897500000D-01	6.046488770D+00	9.989950085D-06	7.615583985D-01

THE FINAL ESTIMATE OF BETA = 4.8975000D-01

THE FINAL ESTIMATE OF LAMBDA = 7.6155840D-01

CONVERGENCE TO 9.9899501D-06
WHICH IS LESS THAN EPSILON = 1.0000000D-05

THE FINAL STEP SIZE IS 1.0000000D-05

CRAMER - VON MISES GOODNESS OF FIT TEST.

ESTIMATED BETA = 4.8975000D-01

NUMBER OF FAILURES = 40

UNBIASED ESTIMATE OF BETA = 4.6526478D-01

CRAMER - VON MISES STATISTIC = 6.8282282D-02

REJECT THE WEIBULL INTENSITY MODEL IF THE STATISTIC
EXCEEDS THE APPROPRIATE CRITICAL VALUE FOR M = 39

DATA INPUT PHASE.

INDATA ERROR -- A FAILURE AT 0.0000000 WAS INPUT.

THE PROBABILITY OF SUCH A FAILURE TIME IS 0.0 ACCORDING TO THE MODEL.

TOTAL NUMBER OF FAILURES =   7

TOTAL NUMBER OF SYSTEMS =   1

SYSTEM STARTING AND ENDING TIMES.

                              FAILURES.

   0.   7.900D+01

                      0.   2.000D+01 4.500D+01 6.200D+01 6.400D+01
                      7.900D+01 6.800D+01

DATA INPUT PHASE.

TOTAL NUMBER OF FAILURES =   5

TOTAL NUMBER OF SYSTEMS =   1

SYSTEM STARTING AND ENDING TIMES.

                                   FAILURES.

    4.560D+02   7.890D+02

                          4.560D+02 4.561D+02 4.570D+02 4.580D+02 4.890D+02

ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY FUNCTION.

CONSTANT NOT INVOLVING BETA: A =   6.1378240D+00

ESTIMATED BETA	FUNCTION O(B')	A - O(B')	ESTIMATED LAMBDA
1.0000000000D+01	6.5730555290+00	-4.3523153700-01	5.3704355370-29
9.0000000000D+00	6.5636282560+00	-4.2580426400-01	4.2502328940-26
8.0000000000D+00	6.5526773070+00	-4.1485331580-01	3.3712739540-23
7.0000000000D+00	6.5399780970+00	-4.0215410580-01	2.6846472360-20
6.0000000000+00	6.5253232290+00	-3.8749923780-01	2.1527927750-17
5.0000000000+00	6.5085569710+00	-3.7073297930-01	1.7479656280-14
4.0000000000D+00	6.4896198020+00	-3.5179581000-01	1.4522429250-11
3.0000000000D+00	6.468596384D+00	-3.3077239200-01	1.2615104850-08
2.0000000000D+00	6.4457540820+00	-3.0793009020-01	1.2060253020-05
1.0000000000D+00	6.4215552730+00	-2.8373128180-01	1.5015015020-02
9.0000000000-01	6.419083888D+00	-2.8125989690-01	3.1704011970-02
8.0000000000D-01	6.4166058560+00	-2.7878186490-01	6.7762626330-02
7.0000000000D-01	6.4141218920+00	-2.7629790050-01	1.4709499980-01
6.0000000000D-01	6.4116327180+00	-2.7380872670-01	3.2587604730-01
5.0000000000D-01	6.4091390650+00	-2.7131507380-01	7.4239189660-01
4.0000000000D-01	6.4066416700+00	-2.6881767870-01	1.7613042810+00
3.0000000000-01	6.4041412750+00	-2.6631728330-01	4.4561084710+00
2.0000000000-01	6.4016386250+00	-2.6381463390-01	1.2680050091D+01
1.0000000000-01	6.3991344720+00	-2.6131048020-01	4.8096807790+01
9.0000000000-02	6.3988840003D+00	-2.6106001100-01	5.6972370190+01
8.0000000000-02	6.3986335270+00	-2.6080953500-01	6.8329189010+01
7.0000000000-02	6.3983830450+00	-2.6055905300-01	8.3250448440+01
6.0000000000-02	6.3981325570+00	-2.6030856570-01	1.0354297810+02
5.0000000000-02	6.397882066D+00	-2.6005807400-01	1.3246101870+02
4.0000000000-02	6.3976315700+00	-2.5980757840-01	1.7651564680+02
3.0000000000-02	6.3973810710+00	-2.5955707990-01	2.5090306230+02
2.0000000000-02	6.3971305710+00	-2.5930657900-01	4.0121776100+02
1.0000000000-02	6.3968800680+00	-2.5905607670-01	8.5544479230+02
9.0000000000D-03	6.3968550180+00	-2.5903102640-01	9.5659388780+02
8.0000000000D-03	6.396829968D+00	-2.5900597610-01	1.0830742670+03
7.0000000000D-03	6.3968049170+00	-2.5898092580-01	1.2457425170+03
6.0000000000-03	6.3967798670+00	-2.5895587550-01	1.4626929500+03
5.0000000000-03	6.3967548170+00	-2.5893082520-01	1.766495336D+03
4.0000000000D-03	6.396729766D+00	-2.5890577490-01	2.2222892110+03
3.0000000000-03	6.3967047160+00	-2.5888072460-01	2.9820668400+03
2.0000000000D-03	6.396679666D+00	-2.5885567430-01	4.5018050160+03
1.0000000000D-03	6.396654616D+00	-2.5883062400-01	9.0613877230+03
9.0000000000-04	6.3966521100+00	-2.5882811890-01	1.0074650930+04
8.0000000000D-04	6.3966499605+00	-2.5882561390-01	1.134234560+04
7.0000000000D-04	6.3966471000+00	-2.5882310890-01	1.296970454D+04
6.0000000000-04	6.3966445595+00	-2.5882060380-01	1.514100403D+04
5.0000000000-04	6.3966420900+00	-2.5881809880-01	1.8180830750+04
4.0000000000-04	6.3966395850+00	-2.5881559380-01	2.2740580110+04
3.0000000000-04	6.3966370800+00	-2.5881308870-01	3.0340174790+04
2.0000000000-04	6.3966345750+00	-2.5881058370-01	4.5539382760+04
1.0000000000-04	6.3966320700+00	-2.5880807870-01	9.1137043900+04
9.0000000000-05	6.3966318200+00	-2.5880782820-01	1.0126985980+05
8.0000000000D-05	6.3966315690+00	-2.5880757770-01	1.1393588010+05
7.0000000000-05	6.3966313190+00	-2.5880732720-01	1.3022076380+05
6.0000000000-05	6.3966310680+00	-2.5880707670-01	1.5193394280+05

5.000000000D-05	6.396630818D+00	-2.588068262D-01	1.823323941D+05
4.000000000D-05	6.396630567D+00	-2.588065756D-01	2.279300721D+05
3.000000000D-05	6.396630317D+00	-2.588063251D-01	3.039262031D+05
2.000000000D-05	6.396630066D+00	-2.588060746D-01	4.559184672D+05
1.000000000D-05	6.396629816D+00	-2.588058241D-01	9.118952631D+05
9.000000000D-06	6.396629791D+00	-2.588057991D-01	1.013223440D+06
8.000000000D-06	6.396629766D+00	-2.588057740D-01	1.139883662D+06
7.000000000D-06	6.396629741D+00	-2.588057490D-01	1.302732518D+06
6.000000000D-06	6.396629715D+00	-2.588057239D-01	1.519864326D+06
5.000000000D-06	6.396629690D+00	-2.588056989D-01	1.823848858D+06
4.000000000D-06	6.396629665D+00	-2.588056738D-01	2.279825655D+06
3.000000000D-06	6.396629640D+00	-2.588056488D-01	3.039786985D+06
2.000000000D-06	6.396629615D+00	-2.588056237D-01	4.559709644D+06
1.000000000D-06	6.396629590D+00	-2.588055987D-01	9.119477621D+06
9.000000000D-07	6.396629588D+00	-2.588055962D-01	1.013275939D+07
8.000000000D-07	6.396629585D+00	-2.588055937D-01	1.139936161D+07
7.000000000D-07	6.396629583D+00	-2.588055912D-01	1.302785017D+07
6.000000000D-07	6.396629580D+00	-2.588055887D-01	1.519916826D+07
5.000000000D-07	6.396629578D+00	-2.588055862D-01	1.823901358D+07
4.000000000D-07	6.396629575D+00	-2.588055837D-01	2.279878155D+07
3.000000000D-07	6.396629573D+00	-2.588055812D-01	3.039839485D+07
2.000000000D-07	6.396629570D+00	-2.588055786D-01	4.559762144D+07
1.000000000D-07	6.396629568D+00	-2.588055761D-01	9.119530122D+07
9.000000000D-08	6.396629567D+00	-2.588055759D-01	1.013281189D+08
8.000000000D-08	6.396629567D+00	-2.588055756D-01	1.139941411D+08
7.000000000D-08	6.396629567D+00	-2.588055754D-01	1.302790267D+08
6.000000000D-08	6.396629567D+00	-2.588055751D-01	1.519922076D+08
5.000000000D-08	6.396629566D+00	-2.588055749D-01	1.823906608D+08
4.000000000D-08	6.396629566D+00	-2.588055746D-01	2.279883406D+08
3.000000000D-08	6.396629566D+00	-2.588055744D-01	3.039844735D+08
2.000000000D-08	6.396629566D+00	-2.588055741D-01	4.559767394D+08
1.000000000D-08	6.396629565D+00	-2.588055739D-01	9.119535372D+08

MLEW ERROR -- BETA LESS THAN 0.00000001

DATA INPUT PHASE.

TOTAL NUMBER OF FAILURES =   12

TOTAL NUMBER OF SYSTEMS =   1

SYSTEM STARTING AND ENDING TIMES.

                                FAILURES.

    0.   8.000D+02
                                4.500D+01 4.560D+02 4.670D+02 4.770D+02 4.840D+02
                                4.920D+02 1.200D-01 5.640D+02 6.500D+01 7.800D+01
                                8.900D+01 9.910D+01

ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY FUNCTION.

CONSTANT NOT INVOLVING BETA: A =   4.7045181D+00

ESTIMATED BETA	FUNCTION D(B')	A - D(B')	ESTIMATED LAMBDA
1.000000000D+01	6.584611728D+00	-1.880093620D+00	1.117587090D-28
9.000000000D+00	6.573500617D+00	-1.868982509D+00	8.940696716D-26
8.000000000D+00	6.559611728D+00	-1.855093620D+00	7.152557373D-23
7.000000000D+00	6.541754585D+00	-1.837236477D+00	5.722045898D-20
6.000000000D+00	6.517945061D+00	-1.813426953D+00	4.577636719D-17
5.000000000D+00	6.484611728D+00	-1.780093620D+00	3.662109375D-14
4.000000000D+00	6.434611728D+00	-1.730093620D+00	2.929687500D-11
3.000000000D+00	6.351278394D+00	-1.646760287D+00	2.343750000D-08
2.000000000D+00	6.184611728D+00	-1.480093620D+00	1.875000000D-05
1.000000000D+00	5.684611728D+00	-9.800936198D-01	1.500000000D-02
9.000000000D-01	5.573500617D+00	-8.689825087D-01	2.926848599D-02
8.000000000D-01	5.434611728D+00	-7.300936198D-01	5.709618160D-02
7.000000000D-01	5.256040299D+00	-5.515221913D-01	1.114341373D-01
6.000000000D-01	5.017945061D+00	-3.134269532D-01	2.174338991D-01
5.000000000D-01	4.684611728D+00	1.990638017D-02	4.242640687D-01
5.900000000D-01	4.989696473D+00	-2.851783656D-01	2.324653100D-01
5.800000000D-01	4.960473797D+00	-2.559556888D-01	2.485358565D-01
5.700000000D-01	4.930225763D+00	-2.257076549D-01	2.657173750D-01
5.600000000D-01	4.898897442D+00	-1.943793341D-01	2.840866682D-01
5.500000000D-01	4.866429909D+00	-1.619118017D-01	3.037258479D-01
5.400000000D-01	4.832759876D+00	-1.282417680D-01	3.247227028D-01
5.300000000D-01	4.797819275D+00	-9.330116700D-02	3.471710900D-01
5.200000000D-01	4.761534805D+00	-5.701669676D-02	3.711713554D-01
5.100000000D-01	4.723827414D+00	-1.930930611D-02	3.968307819D-01
5.000000000D-01	4.684611728D+00	1.990638017D-02	4.242640687D-01
5.090000000D-01	4.719975185D+00	-1.545707759D-02	3.994923273D-01
5.080000000D-01	4.716107791D+00	-1.158968283D-02	4.021717238D-01
5.070000000D-01	4.712225140D+00	-7.707032063D-03	4.048690910D-01
5.060000000D-01	4.708327143D+00	-3.809034854D-03	4.075845495D-01
5.050000000D-01	4.704413708D+00	1.043999676D-04	4.103182205D-01
5.059000000D-01	4.707936495D+00	-3.418387575D-03	4.078570950D-01
5.058000000D-01	4.707545694D+00	-3.027585829D-03	4.081298228D-01
5.057000000D-01	4.707154737D+00	-2.636629524D-03	4.084027329D-01
5.056000000D-01	4.706763626D+00	-2.245518569D-03	4.086758256D-01
5.055000000D-01	4.706372361D+00	-1.854252871D-03	4.089491008D-01
5.054000000D-01	4.705980940D+00	-1.462832339D-03	4.092225588D-01
5.053000000D-01	4.705589365D+00	-1.071256882D-03	4.094961996D-01
5.052000000D-01	4.705197634D+00	-6.795264061D-04	4.097700235D-01
5.051000000D-01	4.704805749D+00	-2.876408204D-04	4.100440304D-01
5.050000000D-01	4.704413708D+00	1.043999676D-04	4.103182205D-01
5.050900000D-01	4.704766552D+00	-2.484437272D-04	4.100714412D-01
5.050800000D-01	4.704727353D+00	-2.092450819D-04	4.100988538D-01
5.050700000D-01	4.704688153D+00	-1.700448844D-04	4.101262682D-01
5.050600000D-01	4.704648951D+00	-1.308431346D-04	4.101536845D-01
5.050500000D-01	4.704609748D+00	-9.163983243D-05	4.101811026D-01
5.050400000D-01	4.704570543D+00	-5.243497777D-05	4.102085225D-01
5.050300000D-01	4.704531336D+00	-1.322857053D-05	4.102359442D-01
5.050200000D-01	4.704492128D+00	2.597938938D-05	4.102633678D-01
5.050290000D-01	4.704527416D+00	-9.307844409D-06	4.102386865D-01

THE FINAL ESTIMATE OF BETA = 5.0502900D-01

THE FINAL ESTIMATE OF LAMBDA = 4.1023869D-01

CONVERGENCE TO 9.3078444D-06
WHICH IS LESS THAN EPSILON = 1.0000000D-05

THE FINAL STEP SIZE IS 1.0000000D-06

CRAMER - VON MISES GOODNESS OF FIT TEST.

ESTIMATED BETA = 5.0502900D-01

NUMBER OF FAILURES = 12

UNBIASED ESTIMATE OF BETA = 4.6294107D-01

CRAMER - VON MISES STATISTIC = 1.4645334D-01

REJECT THE WEIBULL INTENSITY MODEL IF THE STATISTIC
EXCEEDS THE APPROPRIATE CRITICAL VALUE FOR M = 12

DATA INPUT PHASE.

TOTAL NUMBER OF FAILURES = 13

TOTAL NUMBER OF SYSTEMS = 4

SYSTEM STARTING AND ENDING TIMES.

<div style="text-align:center">FAILURES.</div>

0.   1.000D+01

2.000D+00 3.000D+00 4.000D+00 5.000D+00 6.000D+00
8.000D+00

0.   3.000D+01

NO FAILURES

0.   5.000D+01

4.100D+01 4.200D+01 4.300D+01 4.400D+01

0.   6.000D+01

5.100D+01 5.200D+01 5.300D+01

67

ESTIMATION OF THE PARAMETERS OF THE WEIBULL INTENSITY FUNCTION.

CONSTANT NOT INVOLVING BETA: A =   2.7314373D+00

ESTIMATED BETA	FUNCTION D(B')	A - D(B')	ESTIMATED LAMBDA
1.000000000D+01	3.968432009D+00	-1.236994667D+00	1.849458331D-17
9.000000000D+00	3.952550770D+00	-1.221113428D+00	1.078792963D-15
8.000000000D+00	3.932861136D+00	-1.201423794D+00	6.259621158D-14
7.000000000D+00	3.907735525D+00	-1.176298182D+00	3.608615152D-12
6.000000000D+00	3.874419411D+00	-1.142982068D+00	2.063131834D-10
5.000000000D+00	3.827948744D+00	-1.096511402D+00	1.166443351D-08
4.000000000D+00	3.758529458D+00	-1.027092115D+00	6.490264603D-07
3.000000000D+00	3.643675381D+00	-9.122380383D-01	3.523035230D-05
2.000000000D+00	3.417047055D+00	-6.856097124D-01	1.830985915D-03
1.000000000D+00	2.775490643D+00	-4.405330001D-02	8.666666667D-02
9.000000000D-01	2.641348518D+00	9.008882500D-02	1.262779044D-01
9.900000000D-01	2.763183818D+00	-3.174647579D-02	9.000032102D-02
9.800000000D-01	2.750649704D+00	-1.921236129D-02	9.346013376D-02
9.700000000D-01	2.737881746D+00	-6.444403620D-03	9.705077775D-02
9.600000000D-01	2.724873127D+00	6.564215663D-03	1.007770941D-01
9.690000000D-01	2.736591829D+00	-5.154486343D-03	9.741722276D-02
9.680000000D-01	2.735299498D+00	-3.862155711D-03	9.778502929D-02
9.670000000D-01	2.734004747D+00	-2.567404728D-03	9.815420229D-02
9.660000000D-01	2.732707569D+00	-1.270226370D-03	9.852474672D-02
9.650000000D-01	2.731407956D+00	2.938641763D-05	9.889666758D-02
9.659000000D-01	2.732577717D+00	-1.140374757D-03	9.856187678D-02
9.658000000D-01	2.732447841D+00	-1.010498793D-03	9.859902062D-02
9.657000000D-01	2.732317941D+00	-8.805984704D-04	9.863617822D-02
9.656000000D-01	2.732188016D+00	-7.506737824D-04	9.867334961D-02
9.655000000D-01	2.732058067D+00	-6.207247218D-04	9.871053478D-02
9.654000000D-01	2.731928094D+00	-4.907512815D-04	9.874773374D-02
9.653000000D-01	2.731798096D+00	-3.607534545D-04	9.878494650D-02
9.652000000D-01	2.731668074D+00	-2.307312337D-04	9.882217305D-02
9.651000000D-01	2.731538027D+00	-1.006846121D-04	9.885941341D-02
9.650000000D-01	2.731407956D+00	2.938641763D-05	9.889666758D-02
9.650900000D-01	2.731525021D+00	-8.767860756D-05	9.886313820D-02
9.650800000D-01	2.731512015D+00	-7.467235898D-05	9.886686314D-02
9.650700000D-01	2.731499008D+00	-6.165866631D-05	9.887058821D-02
9.650600000D-01	2.731486002D+00	-4.865912954D-05	9.887431342D-02
9.650500000D-01	2.731472995D+00	-3.565214866D-05	9.887803877D-02
9.650400000D-01	2.731459987D+00	-2.264492366D-05	9.888176425D-02
9.650300000D-01	2.731446980D+00	-9.637454539D-06	9.888548988D-02

THE FINAL ESTIMATE OF BETA = 9.6503000D-01

THE FINAL ESTIMATE OF LAMBDA = 9.8885490D-02

CONVERGENCE TO 9.6374545D-06
WHICH IS LESS THAN EPSILON = 1.0000000D-05

THE FINAL STEP SIZE IS 1.0000000D-05

CRAMER - VON MISES GOODNESS OF FIT TEST.

ESTIMATED BETA = 9.6503000D-01

NUMBER OF FAILURES = 13

UNBIASED ESTIMATE OF BETA = 1.9237087D+00

CRAMER - VON MISES STATISTIC = 1.6179976D-01

REJECT THE WEIBULL INTENSITY MODEL IF THE STATISTIC
EXCEEDS THE APPROPRIATE CRITICAL VALUE FOR M = 13

DATA INPUT PHASE.

PROGRAM RUN ENDS NORMALLY.

Next page is blank.

DISTRIBUTION LIST

No. of Copies	Organization
12	Commander Defense Documentation Center ATTN: TCA Cameron Station Alexandria, VA 22314
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCCP 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDE-F 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCRE-I 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCBSI-L 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCPA-S 5001 Eisenhower Avenue Alexandria, VA 22333

DISTRIBUTION LIST

No. of Copies	Organization
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCQA 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCBSI-D 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDE-R 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDE-D 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-SA Rock Island, IL 61299
1	Commander Rock Island Arsenal ATTN: Tech Lib Rock Island, IL 61299
1	Commander US Army Armament Research and Development Command Dover, NJ 07801

DISTRIBUTION LIST

No. of Copies	Organization
1	Commander Harry Diamond Laboratories ATTN: DELHD-SAB 2800 Powder Mill Road Adelphi, MD 20783
1	Commander US Army Test & Evaluation Command ATTN: STEDP-MT-L Dugway Proving Ground, MD 84022
1	Commander US Army Aviation Research and Development Command ATTN: DRDAV-BC P. O. Box 209 St. Louis, MO 63166
1	Commander US Army Electronics Research and Development Command ATTN: DRDEL-SA Fort Monmouth, NJ 07703
2	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL ATAA-T White Sands Missile Range, NM 88002
1	Commander US Army Missile Command ATTN: DRSMI-C Redstone Arsenal, AL 35809
1	Commander US Army Troop Support and Aviation Materiel Readiness Command ATTN: DRSTS-F 4300 Goodfellow Blvd St. Louis, MO 63120

DISTRIBUTION LIST

No. of Copies	Organization
1	Commander US Army Tank-Automotive Research and Development Command ATTN: DRDTA-UL (Tech Lib) DRDTA-V Warren, MI 48090
1	Commander US Army Mobility Equipment Research and Development Command ATTN: DRDME-O Fort Belvoir, VA 22060
1	Commander US Army Natick Research and Development Command ATTN: DRDNA-O Natick, MA 01760
2	Chief Defense Logistics Studies Information Exchange US Army Logistics Management Center ATTN: DRXMC-D Fort Lee, VA 23801
1	Commander US Army Concepts Analysis Agency 8120 Woodmont Avenue Bethesda, MD 20014
1	Commander US Army Logistics Center ATTN: ATCL-MER (Mr. H. Burnette) Fort Lee, VA 23801

Aberdeen Proving Ground

Cdr, USATECOM
ATTN: DRSTE, DRSTE-CS-A
Bldg 314

Dir, BRL, ATTN: STINFO BR, Bldg 305

Director, BRL, Bldg 328

Director, HEL, Bldg 520